On-Demand Secure Teleconferencing on Public Cloud Infrastructures

Bernardo Pericacho Sánchez

MÁSTER EN INVESTIGACIÓN EN INFORMÁTICA. FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

Trabajo Fin Máster en Arquitectura de Computadores

Curso 2012/2013

Director/es y/o colaborador:

Ignacio Martín Llorente
José Luis Vázquez-Poletti

Calificación: SB

Autorización de difusión

Bernardo Pericacho Sánchez

20 de junio de 2013

El/la abajo firmante, matriculado/a en el Máster en Investigación en Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Máster: "On-Demand Secure Teleconferencing on Public Cloud Infrastructures", realizado durante el curso académico 2012-2013 bajo la dirección de Ignacio Martín Llorente y José Luis Vázquez-Poletti en el Departamento de Arquitectura de Computadores y Automática, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.

Resumen en castellano

Uno de los principales obstáculos de la migración de las aplicaciones al cloud es la seguridad. No tener ningún control sobre cómo viajan nuestros datos a través de Internet y no conocer quién tiene acceso a ellos, hace que los usuarios sean reacios a adoptar una estrategia de migración de aplicaciones a cloud públicos. Un ejemplo concreto son las comunicaciones vía Internet, donde los usuarios pueden requerir comunicarse de una forma segura y no susceptible de espionaje o interceptación, es decir, sin tener una tercera persona escuchando la conversación. Este trabajo presenta dos arquitecturas de teleconferencia segura desplegadas en la nube, *VPN Secure Cloud Teleconferencing Architecture* y *Hidden Server Cloud Teleconferencing Architecture*, para abordar las posibles brechas de seguridad y ataques en las comunicaciones de voz sobre IP (VoIP) en un medio inseguro como Internet. Además, se propone un modelo representado por un árbol de decisión, en el que un usuario puede determinar de forma sencilla que arquitectura e infraestructura es la que mejor se ajusta a sus necesidades a la hora de establecer un servidor de teleconferencia segura en un cloud público. Este trabajo está organizado en tres partes: en la primera se presenta la motivación científica y el planteamiento del proyecto: el desarollo de arquitecturas capaces de hacer frente a los problemas de seguridad de las comunicaciones vía Internet; en la segunda, se detalla el proceso de despliegue de dichas arquitecturas y sus características; y en la tercera y última se presentan algunas de las conclusiones obtenidas y se identifican futuras líneas de trabajo.

Palabras clave

Computación en la nube, seguridad en las comunicaciones, VoIP

Abstract

One of the main handicaps about migrating applications to the cloud is security. Having no control over how our data travels through the Internet and not knowing who has access to them, make users reluctant to adopt a public cloud migration strategy. Communications over the Internet are a concrete example where users may need to communicate in a secure way not susceptible to eavesdropping or interception, i.e. not having a third party to listen in. In this document, two new secure teleconferencing architectures deployed in the cloud are presented, *Teleconferencing Secure Cloud VPN Architecture* and *Hidden Teleconferencing Server Cloud Architecture*, to address possible security breaches and attacks in voice over IP (VoIP) communications in an unsafe environment such as the Internet. In addition, a model represented by a decision tree is proposed, in which a user can easily determine which architecture and infrastructure is the one that best fits his needs when establishing a secure teleconference server on a public cloud. This paper is organized in three parts: part I presents the scientific motivation and approach to the project: the development of architectures able to address security issues in Internet communications; in part II, the deployment of these architectures and their characteristics are detailed; and in part III, some of the conclusions are presented and future lines of research identified.

Keywords

Cloud computing, communications security, VoIP

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to express my sincerest gratitude to:

- my relatives who instilled into me the ethic and rigor that guide the passage through life.
- the project directors for their outstanding leadership and tireless efforts.

Part I

Motivation and objectives

Chapter 1

Introduction

Before cloud computing emergence, particular responsibilities of executives at large companies included making sure that all of the employees had the right hardware and software they needed to do their jobs. Buying computers for everyone was not enough. They also had to purchase software or software licenses to give employees the tools they required. Now, in the cloud computing era, instead of installing a suite of software for each computer, this executive would only have to load one application. This application would allow workers to log into a Web-based service which hosts all the programs the user would need for his or her job. Remote machines owned by another company would run everything from e-mail to word processing to complex data analysis programs.

In a cloud computing system, there is a significant workload shift. Local computers no longer have to do all the heavy lifting when it comes to running applications. The network of computers that make up the cloud handles them instead. Hardware and software demands on the user's side decrease. The only thing the user's computer needs to be able to run is the cloud computing system's interface software, which can be as simple as a Web browser, and the cloud's network takes care of the rest.

This is not the only benefit of adopting a cloud computing system. Proponents claim that cloud computing allows companies to avoid upfront infrastructure costs, and focus on

projects that differentiate their businesses instead of infrastructure. Proponents also claim that cloud computing allows enterprises to get their applications up and running faster, with improved manageability and less maintenance, and enables IT to more rapidly adjust resources to meet fluctuating and unpredictable business demand. In addition, cloud computing provides the tools and technologies to build data/compute intensive parallel applications with much more affordable prices compared to traditional parallel computing techniques.

Extra benefits of adopting a business perspective cloud-based are: broad network access (capabilities are available over the network) and elasticity (capabilities can be elastically provisioned and released). However, a cloud migration can present numerous challenges such as performance and raise security concerns.

Cloud communications are Internet-based voice and data communications where telecommunications applications, switching and storage are hosted by a third-party outside of the organization using them, and they are accessed over the public Internet. Most traditional communications media including telephone, music, film, and television are being reshaped or redefined by the Internet, giving birth to new services such as voice over Internet Protocol (VoIP). Cloud communications providers deliver voice & data communications applications and services, hosting them on servers that the providers own and maintain, giving their customers access to the "cloud". Because they only pay for services or applications they use, customers have a more cost-effective, reliable and secure communications environment, without the headaches associated with more conventional PBX system deployment. In short, companies can cut costs with cloud communications services without sacrificing features.

In cloud-based communications, security is a big issue that needs to be addressed. When a communication is established, people want to communicate in a way not susceptible to eavesdropping or interception, not having a third party to listen in. Furthermore, communication data travels through the Internet, having no control over how our data travels through the Internet and not knowing who has access to them. Security measures have to be taken to prevent attacks to communications and keep data and conversation confidentiality.

Chapter 2

Previous Work

In the following chapters a general overview about cloud computing and communications is given, emphasising on security and performance issues. First, a preliminary training about the Internet is presented, analysing each of its layers. After that, a brief communication state of the art is shown, focusing on VoIP communications and its security breaches. Also, the most used VoIP software are introduced, comparing their features and proprietary protocols. Then, speech quality measure methods are analysed, highlighting its advantages and disadvantages against the others and studying correlation with subjective methods to establish a speech quality estimator.

Due to communication security issues, possible communication security breaches and attacks, current possible secure solutions and preventions are summarised. Finally, a short introduction of cloud computing is given. Different cloud layers, services and models basic principles are introduced, explaining all benefits about cloud migration and possible drawbacks.

2.1 Internet communication

2.1.1 The Internet

The Internet is a global system of interconnected computer networks that use the standard Internet protocol suite (TCP/IP) to serve several billion users worldwide. It is a network of networks that consists of millions of private, public, academic, business, and government networks, of local to global scope, that are linked by a broad array of electronic, wireless and optical networking technologies. The Internet carries an extensive range of information resources and services, such as the inter-linked hypertext documents of the World Wide Web (WWW) and the infrastructure to support email.

Most traditional communications media including telephone, music, film, and television are being reshaped or redefined by the Internet, giving birth to new services such as voice over Internet Protocol (VoIP) and Internet Protocol television (IPTV). Newspaper, book and other print publishing are adapting to Web site technology, or are reshaped into blogging and web feeds. The Internet has enabled and accelerated new forms of human interactions through instant messaging, Internet forums, and social networking. Online shopping has boomed both for major retail outlets and small artisans and traders. Business-to-business and financial services on the Internet affect supply chains across entire industries.

The Internet has no centralized governance in either technological implementation or policies for access and usage; each constituent network sets its own policies. Only the overreaching definitions of the two principal name spaces in the Internet, the Internet Protocol address space and the Domain Name System, are directed by a maintainer organization, the Internet Corporation for Assigned Names and Numbers (ICANN). The technical underpinning and standardization of the core protocols (IPv4 and IPv6) is an activity of the Internet Engineering Task Force (IETF), a non-profit organization of loosely affiliated international participants that anyone may associate with by contributing technical expertise.

The Internet Protocol Suite is the networking model and a set of communications protocols used for the Internet and similar networks. It is commonly known as TCP/IP, because its most important protocols, the Transmission Control Protocol (TCP) and the Internet Protocol (IP) were the first networking protocols defined in this standard.

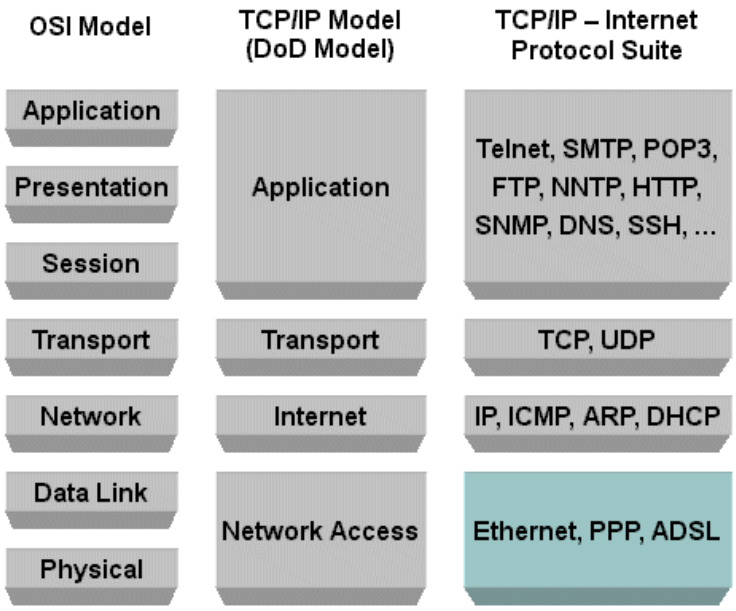

Figure 2.1: *Equivalence of TCP/IP model to OSI model. It is also illustrated some of the most used protocols according to each layer.*

The Internet protocol suite provides end-to-end connectivity specifying how data should be formatted, addressed, transmitted, routed and received at the destination. These restrictions and formats where stipulated in RFC[1] 1122 (Requirements for Internet Hosts – Communication Layers, R. Braden (ed.), October 1989) and RFC 1123 (Requirements for Internet Hosts – Application and Support, R. Braden (ed.), October 1989)[9][10]. It has

[1] A Request for Comments (RFC) is a publication of the Internet Engineering Task Force (IETF) and the Internet Society, the principal technical development and standards-setting bodies for the Internet. An RFC is authored by engineers and computer scientists in the form of a memorandum describing methods, behaviors, research, or innovations applicable to the working of the Internet and Internet-connected systems. It is submitted either for peer review or simply to convey new concepts, information, or (occasionally) engineering humor. The IETF adopts some of the proposals published as RFCs as Internet standards.

four abstraction layers [Figure 2.1] which are used to sort all related protocols according to the scope of networking involved. This abstraction also allows upper layers to provide services that the lower layers do not provide These layers are defined from lowest to highest as follows:

- **Network Access (Link Layer)**: This layer defines the networking methods within the scope of the local network link on which hosts communicate without intervening routers. This layer describes the protocols used to describe the local network topology and the interfaces needed to effect transmission of Internet layer datagrams to next-neighbour hosts. The TCP/IP model's link layer corresponds to the Open Systems Interconnection[2] (OSI) model physical and data link layers, layers one and two of the OSI model [Figure 2.1].

- **Internet Layer**: The internet layer has the task of exchanging datagrams across network boundaries. It is therefore also referred to as the layer that establishes internetworking, indeed, it defines and establishes the Internet. This layer defines the addressing and routing structures used for the TCP/IP protocol suite. The primary protocol in this scope is the Internet Protocol, which defines IP addresses. Its function in routing is to transport datagrams to the next IP router that has the connectivity to a network closer to the final data destination.

- **Transport Layer (host-to-host)**: The transport layer constitutes the networking regime between two network hosts, either on the local network or on remote networks separated by routers. The transport layer provides a uniform networking interface that hides the actual topology (layout) of the underlying network connections. This

[2]The Open Systems Interconnection (OSI) model (ISO/IEC 7498-1) is a conceptual model that characterizes and standardizes the internal functions of a communications system by partitioning it into abstraction layers. The model is a product of the Open Systems Interconnection project at the International Organization for Standardization (ISO). The model groups similar communication functions into one of seven logical layers. A layer serves the layer above it and is served by the layer below it. Two instances at one layer are connected by a horizontal connection on that layer. These seven layers are: *Physical Layer*, *Data Link Layer*, *Network Layer*, *Transport Layer*, *Session Layer*, *Presentation Layer* and *Application Layer*

is where flow-control, error-correction, and connection protocols exist, such as TCP. This layer deals with opening and maintaining connections between Internet hosts.

- **Application Layer (process-to-process)**: This is the scope within which applications create user data and communicate this data to other processes or applications on another or the same host. The communications partners are often called peers. This is where the higher level protocols such as SMTP, FTP, SSH, HTTP, etc. operate.

2.1.2 Communication

Communication is the process of sharing ideas, information and messages with others. The most basic communication methods that are known to man are speech and non-verbal expressions such as facial expressions and body language. Apart from these basic methods of communication there are other methods of communication. These methods began to evolve and become complex as the wants and needs of human beings became complex.

Electronic communication is what expanded the horizons of communication and really boosted the communication industry. This path was shown to us by the founder of the telephone "Alexander Graham Bell". The telephone was the first step towards all the modern methods of communication such as telephone calls, electronic mail or e-mail, satellite broadcasting, cable television and the internet. The Internet has become the number one source of information and communication through out the world. You can see what is happening while it's happening. The Internet is linked to almost every major organization, business and government. Miss-communication can cause catastrophic outcomes especially when it comes to global marketing, finance, economics, trade and war. Today communication has reached a multi-sensory level where any and every thing is possible. The simple act of exchanging ideas has become one of the most important parts of human life.

Internet or Online Communication is a form of communication, using several channels available on the Internet to communicate and interact online to relay a message to a targeted audience.

As we have already introduced, communication is the most popular use of the Internet, with email topping the list of all the technologies used. Although most of the technologies that are unique to the Internet communication are done in text, there is also Internet telephony. Internet telephony is traditional telephone-like communication conducted via the Internet. There is computer-to-computer and computer-to-phone communication. To use the Internet for such communication a user needs to have a microphone, a sound card and speakers.

Chat rooms are communications channels that permit users to write to one another, "chat," on a particular topic, usually using a specific user name to be identified. There are also instant messaging services for communication with people specified and approved by the user. At the beginning of the 21st century many people use cellular or satellite services to connect to the Internet wirelessly. A number of web-enabled devices are used for wireless communication such as smartphones, tablets and others.

Internet communications rapidly became more varied and convenient for users as technology advanced into the 21st century. The types of Internet communication include social networking sites, where members are able to send messages, links, comments, pictures and articles to other members of the site. Communication is not limited to the sender and receiver, but other members who have access to the site of one user can also make comments. This type of relationship is called an "Internet community".

Blog and vlog are another type of Internet communication. Blogging is when a person expresses his or her thoughts, ideas, as well as social and political views, online. Readers of a blogger's writings can make comments and send their links to other readers. Some bloggers build a devout group of readers. A vlog is also for sharing thoughts and ideas online, using digitally recorded sound files. A user records a vlog on some kind of a digital device and then uploads it onto the computer to share with viewers.

Online forums and message boards provide users with the opportunity to express their knowledge or views of a particular subject. Topics for discussion range from sports teams, political agendas, through home improvement to medical experience and procedures. Customers of online retail stores are able to read reviews by other buyers before purchasing an item and also to rate purchased items. In addition, readers of online news articles can comment and communicate with other readers as well as with news reporters and editors through their comments.

These types of communication are not unique to networked computer environments, but due to much improved accessibility on the Internet, they quickly gained popularity. Later they have also become accessible through mobile devices with Internet connection. Some technologies, such as video and audio conferencing and Internet telephony, require more multimedia capabilities of computer systems.

Several issues related to ethical and legal considerations arise from using the Internet for communication. The manner in which communication is implemented on the Internet makes it susceptible to monitoring. We can't assume that communications are private. Some believe that sending email is like sending a message on a postcard. Some laws have been enacted to help protect privacy during electronic communications. These, however, have been difficult to enforce and are rarely applied. One way to protect privacy is to encrypt or

code a message. A common way of encrypting messages is through the use of public and private keys. Although software for encryption is readily available, current policies and laws prohibit its export.

2.1.3 Voice over IP (VoIP)

One of the main kinds of communication through the Internet is Internet telephony. Internet telephony consists of a combination of hardware and software that enables you to use the Internet as the transmission medium of communications services (voice, fax, SMS, voice-messaging) over the public Internet, rather than via the public switched telephone network (PSTN). In its simplest form, PC-to-PC Internet telephony can be as easy as hooking up a microphone to your computer and sending your voice through a cable modem to a person who has Internet telephony software that is compatible with yours. However, this basic form of Internet telephony is not without its problems. Connecting this way is slower than using a traditional telephone, and the quality of the voice transmissions is also not near the quality you would get when placing a regular phone call.

Voice over IP (voice over Internet Protocol, VoIP) is a methodology and group of technologies for the delivery of voice communications and multimedia sessions over Internet Protocol (IP) networks, such as the Internet[1]. The steps and principles involved in originating a VoIP telephone calls are similar to traditional digital telephony, and involve signaling, channel setup, digitization of the analog voice signals, and encoding. Instead of being transmitted over a circuit-switched network, however, the digital information is packetized and transmission occurs as Internet Protocol (IP) packets over a packet-switched network. Such transmission entails careful considerations about resource management different from time-division multiplexing (TDM) networks.

VoIP systems employ session control and signaling protocols to control the signaling, set-up, and tear-down of calls. They transport audio streams over IP networks using special media delivery protocols that encode voice, audio, video with audio codecs and video codecs as Digital audio by streaming media. Various codecs exist that optimize the media stream based on application requirements and network bandwidth; some implementations rely on narrowband and compressed speech, while others support high fidelity stereo codecs. Some popular codecs included are μ-law and a-law versions of G.711, G.722 which is a high-fidelity codec marketed as HD Voice by Polycom, a popular open source voice codec known as iLBC, a codec that only uses 8 kbit/s each way called G.729, G.726, GSM, etc. Nowadays, VoIP is available on many smartphones, personal computers, and on Internet access devices where calls and SMS text messages may be sent over 3G or Wi-Fi.

The main organisms that define Internet and Telecommunication standards are the Internet Engineering Task Force (IETF) and the International Telecommunications Union (ITU), establishing the VoIP rules and protocols.

Voice over IP has been implemented in various ways using both proprietary protocols, as well as protocols based on open standards. Some of the most used protocols are described as follows:

- **H.323**: This is an ITU-T's (International Telecommunications Union) standard that vendors should comply while providing Voice over IP service. This recommendation provides the technical requirements for voice communication over LANs while assuming that no Quality of Service (QoS) is being provided by LANs. It was originally developed for multimedia conferencing on LANs, but was later extended to cover Voice over IP. The first version was released in 1996 while the second version of H.323 came into effect in January 1998. The standard encompasses both point to point com-

	H.323	SIP	MGCP/H.248/Megaco
Standards body	ITU-T	IETF	MGCP/Megaco: IETF H.248: ITU-T
Architecture	Distributed	Distributed, Peer-to-Peer	Centralized
Call Control	Gatekeeper	Proxy/Redirect Server	Call agent/Media Control Gateway / Softswitch
Endpoints	Gateway, terminal	User agent	Media Gateway
Signaling Transport	TCP/UDP	TCP/UDP	MGCP:UDP H.248/Megaco: TCP/UDP
Multimedia	Yes	Yes	Yes
DTMF-relay transport	RTP (Real Time Transport Protocol)	RTP (Real Time Transport Protocol)	RTP (Real Time Transport Protocol)
Fax-relay transport	T.38	T.38	T.38
Supplemental services	By endpoints or call control	By endpoints or call control	By call agent

Table 2.1: *VoIP Standard Protocols.*

munications and multipoint conferences. The products and applications of different vendors can interoperate if they abide by the H.323 specification. H.323 defines four logical components viz., Terminals, Gateways, Gatekeepers and Multipoint Control Units (MCUs). Terminals, gateways and MCUs are known as endpoints.

- **SIP**: This is an IETF's standard for establishing VOIP connections. It is an application layer control protocol for creating,modifying and terminating sessions with one or more participants. The architecture of SIP is similar to that of HTTP (client-server protocol). Requests are generated by the client and sent to the server. The server processes the requests and then sends a response to the client. A request and the responses for that request make a transaction. SIP has INVITE and ACK messages which define the process of opening a reliable channel over which call control messages may be passed. SIP makes minimal assumptions about the underlying transport protocol. This protocol itself provides reliability and does not depend on TCP for

reliability. SIP depends on the Session Description Protocol (SDP) for carrying out the negotiation for codec identification. SIP supports session descriptions that allow participants to agree on a set of compatible media types. It also supports user mobility by proxying and redirecting requests to the user's current location. Services that SIP provides are:

  - *User Location*: determination of the end system to be used for communication.
  - *Call Set-up*: ringing and establishing call parameters at both called and calling party.
  - *User Availability*: determination of the willingness of the called party to engage in communications.
  - *User Capabilities*: determination of the media and media parameters to be used.
  - *Call handling*: the transfer and termination of calls.

- **Media Gateway Control Protocol (MGCP)**: It is another IETF's standard used for controlling telephony gateways from external call control elements called media gateway controllers or call agents. A telephony gateway is a network element that provides conversion between the audio signals carried on telephone circuits and data packets carried over the Internet or over other packet networks. MGCP assumes a call control architecture where the call control intelligence is outside the gateways and handled by external call control elements. The MGCP assumes that these call control elements, or Call Agents, will synchronize with each other to send coherent commands to the gateways under their control. MGCP is, in essence, a master/slave protocol, where the gateways are expected to execute commands sent by the Call Agents.The MGCP implements the media gateway control interface as a set of transactions. The transactions are composed of a command and a mandatory response.

- **Megaco/H.248**: H.248 or Megaco or Gateway Control Protocol is a recommendation from ITU Telecommunication Standardization Sector (ITU-T) which defines protocols

that are used between elements of a physically decomposed multimedia gateway. It is an implementation of the Media Gateway Control Protocol Architecture [16]. H.248 is also called Megaco in IETF domain. It is now known as Gateway Control Protocol. The current standard published in September 2005 by ITU-T is H.248.1: Gateway control protocol: Version 3. H.248/Megaco is standard protocol for controlling the elements of a physically decomposed multimedia gateway, which enables separation of call control from media conversion. H.248/Megaco is a master/slave protocol used to separate the call control logic from the media processing logic in a gateway.

Communication on the IP network is perceived less reliable in contrast to the circuit-switched public telephone network, as it does not provide a network-based mechanism to ensure that data packets are not lost, and are delivered in sequential order. As IP was designed for carrying data, so it does not provide real time guarantees but only provides best effort service. For voice communications over IP to become acceptable to the users, the delay needs to be less than a threshold value and the IETF (Internet Engineering Task Force) is working on this aspect. To ensure good quality of voice, we can use either Echo Cancellation, Packet Prioritization (giving higher priority to voice packets) or Forward Error Correction. Thus, QoS (Quality of Service) is a major issue in VOIP implementations. The issue is how to guarantee that packet traffic for a voice or other media connection will not be delayed or dropped due interference from other lower priority traffic. Problems to consider are:

- **Latency**: Delay for packet delivery.
- **Jitter**: Variations in delay of packet delivery.
- **Packet loss**: Too much traffic in the network causes the network to drop packets.
- **Burstiness of Loss and Jitter**: Loss and Discards (due to jitter) tend to occur in bursts.

For the end user, large delays are burdensome and can cause bad echos. It is hard to have a working conversation with too large delays. You keep interrupting each other. Jitter causes strange sound effects, but can be handled to some degree with "jitter buffers" in the software. Packet loss causes interrupts. Some degree of packet loss won't be noticeable, but lots of packet loss will make sound lousy.

ITU-T and IETF determine requirements to each point mentioned above in order to guarantee a minimum quality of service.

In case of latency, callers usually notice round-trip voice delays of 250 ms or more. ITU-T G.114 recommends a maximum of a 150 ms one-way latency. Since this includes the entire voice path, part of which may be on the public Internet, your own network should have transit latencies of considerably less than 150 ms.

Jitter can be measured in several ways. There are jitter measurement calculations defined in IETF RFC 3550 RTP (A Transport Protocol for Real-Time Applications) and IETF RFC 3611 RTP (Control Protocol Extended Reports - RTCP XR). But, equipment and network vendors often don't detail exactly how they are calculating the values they report for measured jitter. Most VOIP endpoint devices (e.g. VOIP phones and ATAs) have jitter buffers to compensate for network jitter. Jitter buffers (used to compensate for varying delay) further add to the end-to-end delay, and are usually only effective on delay variations less than 100 ms. Jitter must therefore be minimized.

VOIP is not tolerant of packet loss. Even 1% packet loss can "significantly degrade" a VOIP call using a G.711 codec and other more compressing codecs can tolerate even less packet loss.**Reference to CISCO** For example, according to CISCO Systems Inc, the default G.729 codec requires packet loss far less than 1 percent to avoid audible errors.

A number of protocols have been defined to support the reporting of quality of service (QoS) and quality of experience (QoE) for VoIP calls. These include RTCP Extended Report (RFC 3611), SIP RTCP Summary Reports, H.460.9, H.248.30 and MGCP extensions. The RFC 3611 VoIP Metrics block is generated by an IP phone or gateway during a live call and contains information on packet loss rate, packet discard rate (because of jitter), packet loss/discard burst metrics (burst length/density, gap length/density), network delay, end system delay, signal / noise / echo level, Mean Opinion Scores (MOS) and R factors and configuration information related to the jitter buffer.

RFC 3611 VoIP metrics reports are exchanged between IP endpoints on an occasional basis during a call, and an end of call message sent via SIP RTCP Summary Report or one of the other signaling protocol extensions. RFC 3611 VoIP metrics reports are intended to support real time feedback related to QoS problems, the exchange of information between the endpoints for improved call quality calculation and a variety of other applications.

Then, some of the most used and popular VoIP software both proprietary and open source are described. In Figure 2.2 is shown the top VoIP software applications in mobile broadband.

2.1.3.1 Asterisk

Asterisk is an open source framework for building communications applications. Asterisk turns an ordinary computer into a communications server. Asterisk powers IP PBX systems, VoIP gateways, conference servers and other custom solutions. It is used by small businesses, large businesses, call centres, carriers and government agencies, worldwide. Asterisk is free and open source. Like any PBX, it allows attached telephones to make calls to one another, and to connect to other telephone services, such as the public switched telephone network (PSTN) and Voice over Internet Protocol (VoIP) services. The Asterisk project started in

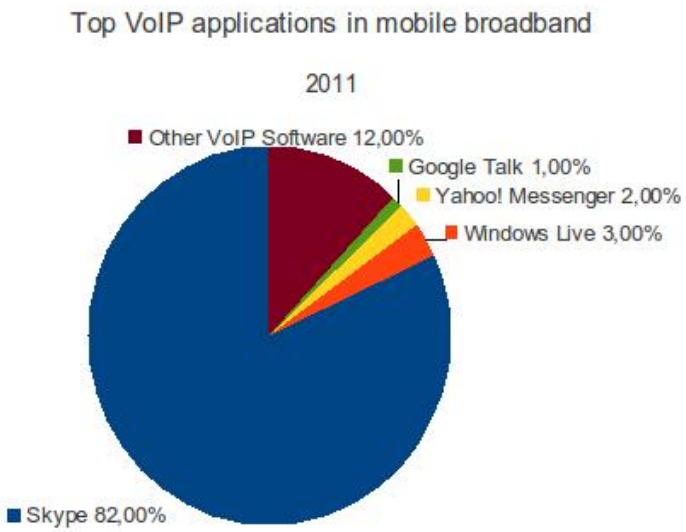

Figure 2.2: *Top VoIP software applications in mobile broadband in 2011.*

1999 when Mark Spencer released the initial code under the GPL open source license. Since that time, it has been enhanced and tested by a global community of thousands. Today, Asterisk is maintained by the combined efforts of Digium and the Asterisk community.

Asterisk is released under a dual license model, using the GNU General Public License (GPL) as a free software license and a proprietary software license to permit licensees to distribute proprietary, unpublished system components. Originally designed for Linux, Asterisk also runs on a variety of different operating systems including NetBSD, OpenBSD, FreeBSD, Mac OS X, and Solaris.

Asterisk supports a wide range of Voice over IP protocols, including the Session Initiation Protocol (SIP), the Media Gateway Control Protocol (MGCP), and H.323. Asterisk can interoperate with most SIP telephones, acting both as registrar and as a gateway between IP phones and the PSTN. The Inter-Asterisk eXchange (IAX2) protocol, RFC 5456, native to Asterisk, provides efficient trunking of calls among Asterisk PBXes, in addition to distributed configuration logic, and call completion to VoIP service providers who support

it. Some telephones support the IAX2 protocol directly. By supporting a mix of traditional and VoIP telephony services, Asterisk allows deployers to build new telephone systems, or gradually migrate existing systems to new technologies. Some sites are using Asterisk servers to replace proprietary PBXes; others to provide additional features (such as voice mail or voice response menus, or virtual call shops) or to reduce costs by carrying long-distance calls over the Internet (toll bypass).

Asterisk was one of the first open source PBX software packages. In addition to VoIP protocols, Asterisk supports many traditional circuit-switching protocols such as ISDN and SS7. This requires appropriate hardware interface cards supporting such protocols, marketed by third-party vendors. Each protocol requires the installation of software modules. With these features, Asterisk provides a wide spectrum of communications options.

2.1.3.2 Skype

Skype is a proprietary Voice over IP service and software application[3]. Skype was first released in 2005 written by Estonian developers Monish Mohan, Priit Kasesalu, and Jaan Tallinn, Danish Janus Friis, and Swedish Niklas Zennström, who had also originally developed Kazaa[4]. It developed into a platform with over 600 million users.

The service allows users to communicate with peers by voice using a microphone, video by using a webcam, and instant messaging over the Internet. Phone calls may be placed to recipients on the traditional telephone networks. Calls to other users within the Skype

[3] www.skype.com

[4] Kazaa initially was a peer-to-peer file sharing application using the FastTrack protocol licensed by Joltid Ltd. and operated as Kazaa by Sharman Networks, but since August 2012, the Kazaa website is no longer active. Kazaa was subsequently under license as a legal music subscription service by Atrinsic, Inc. Kazaa Media Desktop was commonly used to exchange MP3 music files and other file types, such as videos, applications, and documents over the internet. The Kazaa Media Desktop client could be downloaded free of charge; however, it was bundled with adware and for a period there were "No spyware" warnings found on Kazaa's website. During the past few years, Sharman Networks and its business partners and associates were the target of copyright-related lawsuits, related to the copyright of content distributed via Kazaa Media Desktop on the FastTrack protocol.

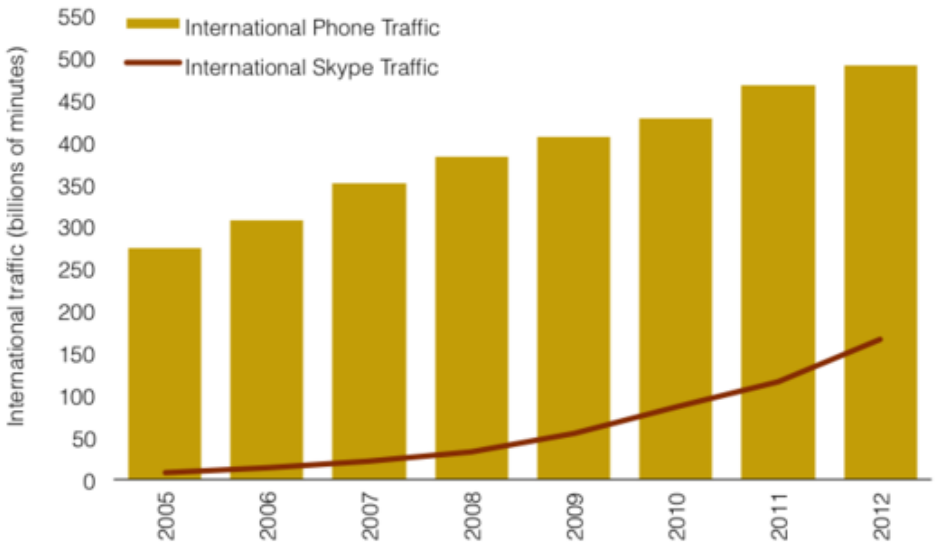

Figure 2.3: *International Telephone and Skype Traffic, 2005-2012.*

service are free of charge, while calls to landline telephones and mobile phones are charged via a debit-based user account system. Skype has also become popular for its additional features, including file transfer, and videoconferencing.

Unlike most other VoIP services, Skype is a hybrid peer-to-peer and client–server system. It makes use of background processing on computers running Skype software. Skype's original proposed name (Sky Peer-to-Peer) reflects this fact.

In January 2011, after the release of video calling on the Skype client for iPhone, Skype reached a record 27 million simultaneous online users. This record was broken with 29 million simultaneous online users on 21 February 2011, and again on 28 March 2011 with 30 million online users. On 25 February 2012, Skype announced that it has over 32 million users for the first time ever. As of 5 March 2012, it has broken to 36 million simultaneous online users and less than a year later, on 21 January 2013, Skype had more than 50 million concurrent users online. In June 2012, Skype had surpassed 70 million downloads on an

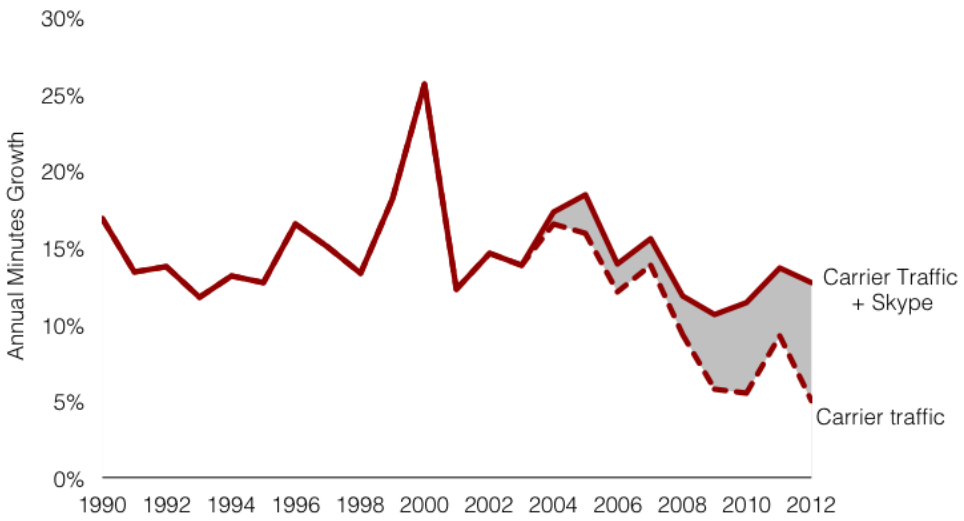

Figure 2.4: *Analysis of the impact of Skype and consumer VoIP services on international calling.*

Android Device. On 19 July 2012, Microsoft announced that Skype users had logged 115 billion minutes of calls over the quarter, up 50% since the last quarter.

Skype uses a proprietary Internet telephony (VoIP) network called the Skype protocol. The protocol has not been made publicly available by Skype and official applications using the protocol are closed-source. Part of the Skype technology relies on the Global Index P2P protocol belonging to the Joltid Ltd. corporation. The main difference between Skype and standard VoIP clients is that Skype operates on a peer-to-peer model (originally based on the Kazaa software), rather than the more usual client–server model (note that the very popular SIP model of VoIP is also peer-to-peer, but implementation generally requires registration with a server, as does Skype). As far as networking stack support, Skype currently only supports the IPv4 protocol and it lacks support for the next generation Internet protocol, IPv6.

Skype uses as audio codecs G.729 and SVOPC. Moreover, Skype added a Skype-created codec called SILK to Skype 4.0 for Windows and other Skype clients, being intended to be "lightweight and embeddable".

Security is the main question to be resolved about Skype. Skype is claimed to be a secure communication.Skype reportedly uses publicly documented, widely trusted encryption techniques: RSA[5] for key negotiation and the Advanced Encryption Standard to encrypt conversations. However, it is impossible to verify that these algorithms are used correctly, completely and at all times as there is no public review possible without a protocol specification and/or the program source code. Skype provides an uncontrolled registration system for users with no proof of identity. Instead, a free choice of nicknames permits users to use the system without revealing their identity to other users. It is trivial to set up an account using any name; the displayed caller's name is no guarantee of authenticity. A third party paper analyzing the security and methodology of Skype was presented at Black Hat Europe 2006 [8]. It analyzed Skype and found a number of security issues with the current security model.

Skype incorporates some features, which tend to hide its traffic, but it is not specifically designed to thwart traffic analysis and therefore does not provide anonymous communication. Some researchers have been able to watermark the traffic so that it is identifiable even after passing through an anonymizing network [46]. For example, in November 2010, a flaw was disclosed to Skype that showed how hackers could secretly track any user's IP address. As of 2013, this still has not been fixed [25].

[5]RSA is an algorithm for public-key cryptography that is based on the presumed difficulty of factoring large integers, the factoring problem. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described the algorithm in 1977. Clifford Cocks, an English mathematician, had developed an equivalent system in 1973, but it was not declassified until 1997. A user of RSA creates and then publishes the product of two large prime numbers, along with an auxiliary value, as their public key. The prime factors must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime factors can feasibly decode the message. Whether breaking RSA encryption is as hard as factoring is an open question known as the RSA problem.

2.1.3.3 Google+ Hangouts

Google+ Hangouts is an instant messaging and video chat platform developed by Google, which launched on May 15, 2013 during the keynote of its I/O development conference. It replaced three messaging products that Google had implemented concurrently within its services, including Talk, Google+ Messenger, and Hangouts, a video chat system present within Google+[6].

Hangouts allows users to hold conversations between two or more users. The service can be accessed online through the Gmail or Google+ websites, or through mobile applications available for Android and iOS (which were distributed as a successor to their existing Google Talk applications). However, because it uses a proprietary protocol instead of the XMPP open standard protocol used by Google Talk, most third-party applications which had access to Google Talk do not have access to Google+ Hangouts. There are, for example, no free software clients for Google+ Hangouts.

One advantage of Google+ Hangouts is that chat histories are saved online, allowing them to be synced between devices. A "watermark" of a user's avatar is used as a marker to indicate how far they have read into the conversation. Photos can be shared during conversations, which are automatically uploaded into a private Google+ album. Users can also now use emoticons symbols in their messages. As with the previous Google+ Hangouts, users can also perform a group video chat with up to 10 users at a time. Nikhyl Singhal, Google's director of real-time communications, stated that its Google Voice service would soon be integrated into Hangouts as well.

[6] http://www.google.com/hangouts/

2.1.3.4 Viber

Viber is a proprietary cross-platform instant messaging voice-over-Internet Protocol application for smartphones developed by Viber Media[7]. In addition to text messaging, users can exchange images, video and audio media messages. The client software is available for Windows, Mac OS, Android, BlackBerry OS, iOS, Series 40, Symbian, Bada, Windows Phone, Mac OS X and Microsoft Windows. Viber works on both 3G and WiFi networks. Viber reached 200 million users as of May 7, 2013.

The actual functionality varies from platform to platform with iOS and Android being the first to receive new features. This is because Viber was initially launched for iPhone on Dec 2 2010, in direct competition with Skype and after that, in July 2012, started releasing non restricted versions for other OS such us Android, Windows Phone and Symbian.

Viber includes text, picture and video messaging across all platforms, with voice calling available only to iPhone, Android and Nokia's Windows Phone although HD voice is planned for Windows Phone 8. The application's user interface includes tab bar on the bottom giving access to messages, recent calls, contact, the keypad and a button for accessing more options. Upon installation, it creates a user account using one's phone number as username. Viber synchronizes with the phone's address book, so users do not need to add contacts in a separate book. Since all users are registered with their phone number, the software returns all Viber users among one's contacts. The newest version of Viber (Version 2.3) has added smileys and other default images.

It has been demonstrated that Viber has many security risks [29]. One of this security issues is that Viber app for iOS and Android devices contains a SQLite Database with some xml files which contain unencrypted configuration data and can be viewed fairly easy.

[7] http://www.viber.com/

2.2 Security in Communications

In general, when a communication is established, we want to communicate in a way not susceptible to eavesdropping or interception, not having a third party to listen in. Communications security is the discipline of preventing unauthorized interceptors from accessing telecommunications in an intelligible form, while still delivering content to the intended recipients.

With many communications taking place over long distance and mediated by technology, and increasing awareness of the importance of interception issues, technology and its compromise are at the heart of this debate.

The VoIP technology is based on the previously threatened IP networks and adds telephony threats as well. As the VoIP technology is evolving, it is collecting vulnerabilities and threats of both Internet and Telecom technologies. The security concept related to VoIP has many different aspects but there are three main fundamentals [Fig. 2.5]: Confidentiality, Integrity and Availability [32].

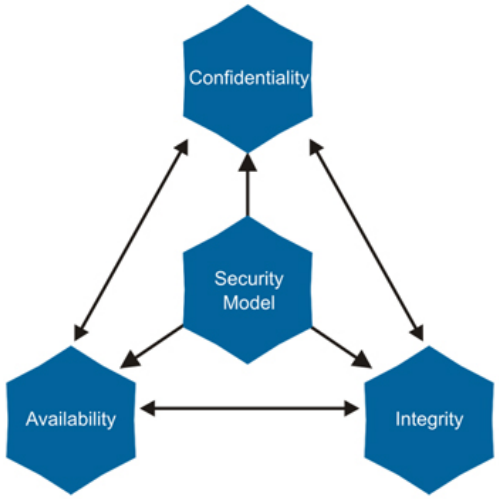

Figure 2.5: *CIA relationship.*

- **Confidentiality**: Confidentiality refers to mechanisms ensuring that only intended recipient have access to the VoIP call. Man-in-the-middle attacks are considered to be confidentiality breaches including eavesdropping, sniffing and application attacks. ARP monitoring encryption and VPN are some techniques to mitigate such attacks.

- **Integrity**: Integrity refers to the prevention of any unauthorized modification in voice packets. Any unauthorized activities must be checked upon. Password breaches are common when a switch reactivates and boots with default settings (**Kuhn et al. 2005**). Further attacks include IP spoofing, quality-degradation, registration/session hijacking and server insertion attacks (**Ramson and Rittinghouse, 2005**). Any rouge packets must be blocked by using VLAN (segmentation), Caller ID verification and fixed routing mechanisms (**Green, 2002**).

- **Availability**: Availability refers that the VoIP services is always available when needed. Denial of Service (DoS) which is a threat to availability could have and adverse effect if the VoIP call centre network if hit by such attack. Other attacks include TCP SYN, SIP INVITE flood and Spam over Internet Telephony (SPIT). Actions needed are using state-full firewalls, Intrusion detection and spam filters on servers.

As introduced, there are many risks associated to VoIP, being classified in attack categories. The technology needs to be secured as the packets take an unspecific route while traversing from source to destination end. Attacks categories are defined as follows:

- **Registration attacks**: These attacks happen where the attacker tries to hack into the system or takes advantage of vulnerabilities in registration injecting themselves into the signal path of the VoIP network. These kind of attacks include IP Spoofing, Theft of Service, Reflection Attack and Brute Force Attack.

- **On call attacks**: These attacks occur when a person is making or receiving a call. The attacker intercepts the route where voice/data packets are being sent. Call Hijacking,

Eavesdropping, ARP spoofing, Connection Hijacking and Signal Protocol Tampering are some of the attacks in this section.

- **Denial of Service attacks**: These type of attacks have no concern about gaining any valuable information. Simply isolates the endpoint of network from the rest of the world by jamming the switches and IP PBX with loads of requests. In this category also are classified attacks such as SIP INVITE Flood, TCP SYN Flood and Malicious RTP Streams.

- **Attacks on VoIP components**: These attacks are primarily on the devices, as they seem to be affected easily. The most common attacks are on the IP PBX, softphones and IP phones.

Due to all kind of attacks that can affect and interfere in VoIP communications it is important to establish a security policy to design a secure VoIP system which can guarantee confidential delivery of services to subscribers.

Virtual Lan is a security technology that allows network administrators to logically divide a LAN into a number un VLANs. This method provides security if any other VLAN is attacked, others remain safe and secure. VLANs use Segmentation which separates voice from data VLAN. In VoIP, two different VLANs for voice and data could be isolated (if there is not a dedicated line for voice), by using a layer three segmentation. All the inter VLAN traffic has to pass through the routing device that filters traffic using access control list.

Virtual Private Network (VPN) is a technology that establishes a private network within the public network. VPN is based on tunnelling, ensuring the provision of confidentiality and integrity of voice packets in IP technology. VPN technique consists of encapsulation. Voice traffic is secured by encapsulating it inside a tunneling standard. The

fundamental mechanism behind tunneling is encryption that ensures confidentiality and data integrity in VoIP networks.

Prior to establishing a connection, tunneling makes use of Authentication Protocol to set up a trust relationship between the network terminal devices. A symmetric encryptions algorithms should be preferred for the voice transportation that would help up in speeding up the process while providing confidentiality. Thus, VPN could use public key cryptography. This method is a good choice to counter man-in-the-middle attacks and similar attacks on voice packets running on VoIP networks.

Mainly there are three subsets of VPN technologies: LAN VPN services, Dial-up VPN services and Extranet VPN Services [40]. The most popular tunneling standards are Point-to-Point Tunneling Protocol (PPTP), Layer 2 Tunneling Protocol and IP Security (IPSec).

The major benefit of VoIP based telephony is the ability to encrypt the digital signals representing the voice stream, being a good defence against call interception. **Encryption** allow to have confidential calls, providing end users the certainty that the message delivered from one end to other regains its secrecy although the voice call is set over a public channel. Thereby, it is needed to prevent attacks from sniffers while making or receiving a call on the public network using encryption-decryption techniques to address confidentiality issue.

As mentioned earlier, latency is an important issue in many converged and real-time services, *symmetric encryption algorithms* are preferred. This algorithm generates a common cryptographic key i.e. shared secret key passed on both sides of the channel [Fig. 2.6]. For example, in IPSec VPN standard, the caller and the callee participating in a voice conversation have to agree previously on a data encryption mechanism included in the standard such as DES, MD5, SHA along with a shared secret key. It should be noted that DES, MD5,

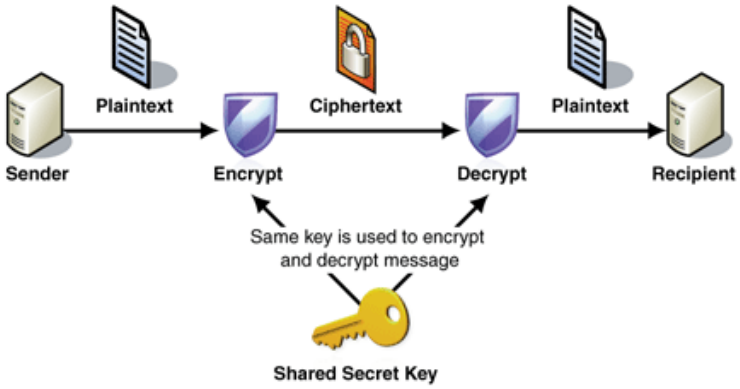

Figure 2.6: *Symmetric encryption algorithm example.*

SHA-0 and SHA-1 have been declared insecure, cryptographically broken and unsuitable for further use [15][2][34].

DES encryption is now consider insecure due to the 56-bit key size being too small. In January, 1999, distributed.net and the Electronic Frontier Foundation collaborated to publicly break a DES key in 22 hours and 15 minutes. There are also some analytical results which demonstrate theoretical weaknesses in the cipher, although they are infeasible to mount in practice. The algorithm is believed to be practically secure in the form of Triple DES, although there are theoretical attacks. In recent years, the cipher has been superseded by the Advanced Encryption Standard (AES). Furthermore, DES has been withdrawn as a standard by the National Institute of Standards and Technology (formerly the National Bureau of Standards).

MD5 encryption is not suitable for applications like SSL certificates or digital signatures that rely on this property, since has been shown that it is not collision resistant[8] [43]. In 1996, a flaw was found with the design of MD5, and while it was not a clearly fatal weakness, cryptographers began recommending the use of other algorithms, such as SHA-1 (which has

[8]Collision resistance is a property of cryptographic hash functions: a hash function is collision resistant if it is hard to find two inputs that hash to the same output; that is, two inputs a and b such that H(a) = H(b), and a $\neq$ b.

since been found to be vulnerable as well). In 2004, more serious flaws were discovered in MD5, making further use of the algorithm for security purposes questionable.

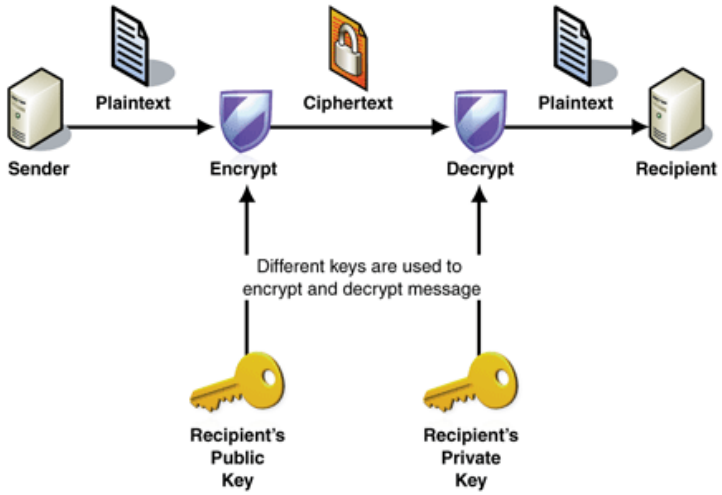

Figure 2.7: *Public key encryption algorithm example.*

If latency is not a big issue, *public key cryptography* could be used as other encryption mechanism. This is regarded as the most secure method. In this scenario, the receiver must obtain the senders public key before any voice connection is established. The sender encrypts the voice/data with his private key, callee must obtain caller's public key before establishing a connection. Caller encrypts the voice call with callee public key and sends it to him. Callee decrypts the voice call and recovers the original voice packets [Fig. 2.7].

In addition to these security technologies, the best to counterattack against registration problems is **authentication**. Is is based on cryptography using common secret or public and private key based methods along with signatures and certificates. In VoIP scenario, a voice conversation which is using the public network could give rise to confidentiality and authentication issues. For example, a person at the end of VoIP conversation could not prove the authenticity of the caller, as caller can decline the authorship of any calls made by him. Thereby, as public keys are widely available, any attacker could intercept the encrypted

data, although he cannot read it but can append any false information or send entirely a new packet encrypted in receivers public key. Ultimately, the problem domain consists in ensuring attackers are not able to masquerade the call and proving the callers and message authentication so that the caller is not able to deny a call made to callee.

Authentication could be provided by different means. One of them is that the subscribers can make use of public and private keys to produce digital signatures technique. The public keys need to be changed before the voice conversation starts. Sender can sign the voice by using his private key and re-encrypt the result with the receiver's public key, thus, only receiver could decrypt the information with his private key and then repeat the same process by using sender's public key. This would provide authentication of the call that has been generated from the legitimate caller.

Another option could be used by applying MD5. The caller takes the hash of voice signal and encrypt the original message and the hash with the callee public key. When the callee receives the call, he decrypts it using his private key and if the hash of the voice signal is equals to the hash received it means the message is original without any modifications.

Both of the above techniques could be applied to make the voice conversation more secure taking into account that the limiting factor is QoS. As follow, it is detailed and open source software that implements an VPN, providing some of the security aspects needed to achieve a secure communication.

2.2.1 OpenVPN

OpenVPN is an open source software application that implements virtual private network (VPN) techniques for creating secure point-to-point or site-to-site connections in routed or bridged configurations and remote access facilities[9]. It uses a custom security protocol that uses SSL/TLS for key exchange. It is capable of traversing network address translators (NATs) and firewalls. It was written by James Yonan and is published under the GNU General Public License (GPL).

OpenVPN allows peers to authenticate each other using a pre-shared secret key, certificates, or username/password. When used in a multiclient-server configuration, it allows the server to release an authentication certificate for every client, using signature and Certificate authority. It uses the OpenSSL encryption library extensively, as well as the SSLv3/TLSv1 protocol, and contains many security and control features.

This open source software application uses the OpenSSL library to provide encryption of both the data and control channels. It lets OpenSSL do all the encryption and authentication work, allowing OpenVPN to use all the ciphers available in the OpenSSL package. It can also use the HMAC packet authentication feature to add an additional layer of security to the connection (referred to as an "HMAC Firewall" by the creator)[30].

OpenVPN can run over User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) transports, multiplexing created SSL tunnels on a single TCP/UDP port. It has the ability to work through most proxy servers (including HTTP) and is good at working through Network address translation (NAT) and getting out through firewalls. The server configuration has the ability to "push" certain network configuration options to the clients. These include IP addresses, routing commands, and a few connection options. OpenVPN

[9] http://openvpn.net/index.php/access-server/docs/admin-guides.html

offers two types of interfaces for networking via the Universal TUN/TAP driver. It can create either a layer-3 based IP tunnel (TUN), or a layer-2 based Ethernet TAP that can carry any type of Ethernet traffic. OpenVPN can optionally use the LZO compression library to compress the data stream. Port 1194 is the official IANA assigned port number for OpenVPN.

As mentioned, OpenVPN has **two authentication modes**: *Static Key* (Use a pre-shared static key) and *TLS* (Use SSL/TLS + certificates for authentication and key exchange).

In static key mode, a pre-shared key is generated and shared between both OpenVPN peers before the tunnel is started. This static key contains 4 independent keys: HMAC send, HMAC receive, encrypt, and decrypt. By default in static key mode, both hosts will use the same HMAC key and the same encrypt/decrypt key. However, using the direction parameter to –secret, it is possible to use all 4 keys independently.

In SSL/TLS mode, an SSL session is established with bidirectional authentication (i.e. each side of the connection must present its own certificate). If the SSL/TLS authentication succeeds, encryption/decryption and HMAC key source material is then randomly generated by OpenSSL's RAND_bytes function and exchanged over the SSL/TLS connection. Both sides of the connection contribute random source material. This mode never uses any key bidirectionally, so each peer has a distinct send HMAC, receive HMAC, packet encrypt, and packet decrypt key. During SSL/TLS rekeying, there is a transition-window parameter that permits overlap between old and new key usage, so there is no time pressure or latency bottleneck during SSL/TLS renegotiations.

OpenVPN can operate on multiple platforms such as Solaris, Linux, OpenBSD, FreeBSD, NetBSD, QNX, Mac OS X, and Windows 2000/XP/Vista/7. Futhermore, it is available for mobile phones OSes like Maemo, Windows Mobile 6.5 and below, iOS 3GS+ devices, jailbroken iOS 3.1.2+ devices and Android 4.0+ devices.

2.3 Speech Quality

Quality of Service (QoS) concept represents a fundamental element in service consumption. QoS are the performance parameter sets values that ensure acceptable quality levels of service to the user. Each kind of service provided has it's own QoS. For example, in traditional telephony, QoS can be defined as having a 64Kbps channel during the conversation time and a availability service of 99,999%. Otherwise, in the case of the Internet, heterogeneous features from the services provided make difficult to identify which are the performance parameters that ensure acceptable quality levels.

Nowadays, QoS parameters used in the Internet are packet loss, delay, jitter and bandwidth among others. All of these parameters affect to voice quality.

There are to many factors that contribute to *point-to-point delay* such as coding algorithm delay, packaging delay, propagation delay (negligible unless large distances), transmission delay, waiting time in the queues of the network (depending on network traffic), decompression time, etc. The total delay from end to end in a voice call should be kept below a certain level to minimize the interactivity loss between users. ITU-T G.114 standard specifies a maximum delay level of 150ms between caller and callee.

Packaged voice transport is not only sensible to point-to-point delay, but also to fluctuations in this time delay (*jitter*). These variations are due to the fluctuations in waiting

Application	Symmetry	Point-to-point delay	Jitter (ms)	Packet loss
Conversation	Round trip	< 150 ms (preferred) < 400 ms (maximum)	< 1 s	$< 3\%$
Audio streaming	One way	< 10 s	< 1 s	$< 1\%$
Voice messages	One way	< 1 s (play) < 2 s (record)	< 1 s	$< 3\%$

Table 2.2: *Maximum values for QoS parameters in voice applications over the Internet specified by ITU-T group 12.*

time in network nodes that depend on the network current traffic. To minimize these jitter effects it is implemented a packet temporal storage (buffers) in the receptor. These buffers also fixed the disordered packet transmission, checking its sequence numbers. However, this technique increases the fixed total point-to-point delay.

Packet loss is another important factor to take into account in real time services. Since the VoIP traffic is implemented under UDP (RTP over UDP), the only control that can be applied is in end points. Codecs implement error correction techniques using interpolation algorithms over the received data to generate loss information. However, when losses exceed certain threshold (around 3%) or when they appear in bursts, these techniques become useless.

In Table 2.2, specified values by ITU-T group 12 for the QoS parameters mentioned above are shown.

As the Internet (IP technology) provides heterogeneous features from the services, we cannot establish threshold values for the QoS parameters to ensure the quality of service. User experience not only depends on network characteristics, but coding, compression and recovery algorithms. Furthermore, the final user (not expert in QoS) may not be interested in packet loss probability or jitter, but needs a perceived quality estimation about the service

at issue. Finally, the quality of service will be for a user what he can perceived of it (PQoS), regardless of the network state.

2.3.1 Perceived Quality of Service (PQoS)

Perceived Quality of Service (PQoS) is the QoS as it is finally perceived by the end-user. Perceived Quality Measures could be made using objective o subjective methods [Fig. 2.8].

Subjective methods define the best accepted metric since represent a direct connection with users perceived quality. These methods consist in evaluate the mean opinion of a group of people, presenting them some sequences and each individual gives a quality value. The inherent problem to this methods is the time needed to accomplished them, high cost and that they cannot be used to monitor quality in large periods of time. These problems have made objective methods attractive to estimate the perceived quality in communication networks.

Objective perceived quality methods can be *intrusive* and *non-intrusive*. Extra signals are necessary to estimate quality in intrusive methods. These methods are more accurate but usually are not suitable to monitor quality of service, due to the necessity of using extra signals and the comparison with the original signals.

Non-intrusive methods do not need extra signals and are suitable to monitor quality of service. Depending on the method's input, they can be classified into signal-based (the input is the transmitted signal) or parameter-based (the inputs are communication network parameters such as bit rate). Objective methods do not give a result in people's opinion, but its result has correlation with the perceived quality. Thus, it is necessary calibration based on the subjective method's results.

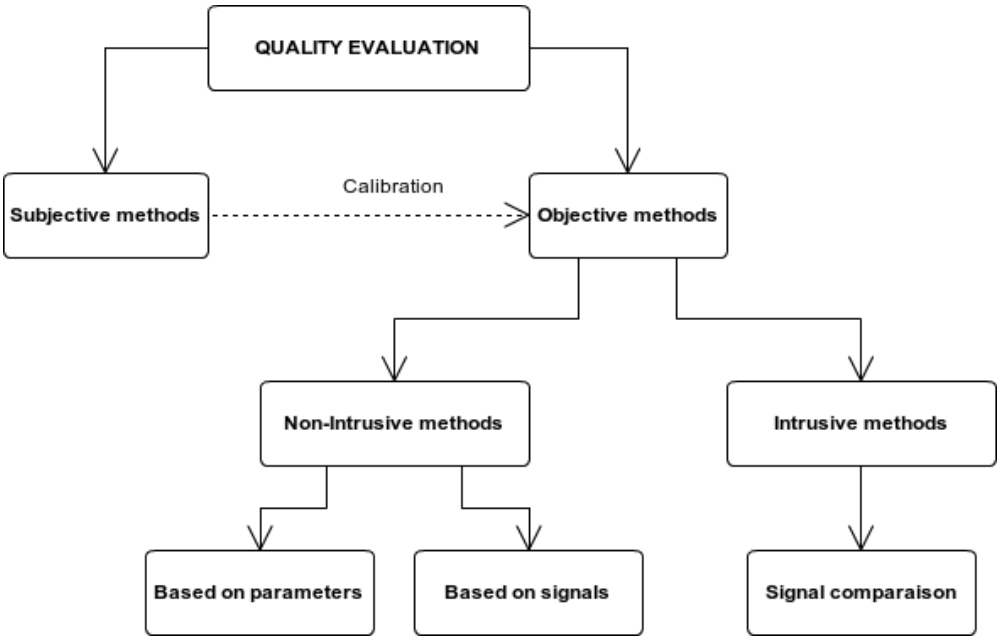

Figure 2.8: *PQoS methods classification.*

As follows, all these methods are detailed, giving a general description of each one.

2.3.1.1 Subjective methods

Different subjective methods are normalize by ITU in ITU P.800 [20] for audio quality. Basically, they could be classified in two kind of test: **Absolute Category Rating (ACR)** resulting *Mean Opinion Score (MOS)* and **Degradation Category Rating (DCR)** resulting *Degradation Mean Opinion Score (DMOS)*. These tests are normally taken in controlled conditions in a laboratory (soundproof rooms).

In ACR test, participants must assign a global quality value to the signal presented, not having access to the original signal. Quality values are assigned within the Table 2.3. The mean value assigned by participants is MOS.

Quality value	Signal Quality
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

Table 2.3: *Absolute Category Rating (ACR) quality values relationship.*

Codec	Bit rate (kbps)	MOS	Compression Delay (ms))
G.711 (PCM)	64	4.1	0.75
G.726 ADPCM	32	3.85	1
G.728 LD-CELP	16	3.61	3-5
G.729 CS-ACELP	8	3.92	10
G.729 x 2 Encodings	8	3.27	10
G.729 x 3 Encodings	8	2.68	10
G729.a CS-ACELP	8	3.7	10
G.723.1 MP-MLQ	6.3	3.9	30
G.723.1 ACELP	5.3	3.65	30

Table 2.4: *Averaged MOS scores for the same sample using different audio codecs.*

In addition, each codec[10] provides a certain quality of speech [Table 2.4][37]. Although it can seem logical from a financial standpoint to convert all calls to low-bit rate codecs to save on infrastructure costs, exercise additional care when you design voice networks with low-bit rate compression. There are drawbacks to compressing voice. One of the main drawbacks is signal distortion due to multiple encodings (called tandem encodings). For example, when a G.729 voice signal is tandem encoded three times, the MOS score drops from 3.92 (very good) to 2.68 (unacceptable). Another drawback is codec-induced delay with low bit-rate codecs.

[10]PCM = Pulse Code Modulation, ADPCM = Adaptive Differential Pulse Code Modulation, LDCELP = Low-Delay Code Excited Linear Prediction, CS-ACELP = Conjugate-Structure Algebraic-Code-Excited Linear-Prediction, MP-MLQ = Multi-Pulse, Multi-Level Quantization, ACELP = Algebraic Code Excited Linear Prediction

Quality value	Degradation Level
5	Imperceptible
4	Perceptible but not annoying
3	Slightly annoying
2	Annoying
1	Very annoying

Table 2.5: *Degradation Category Rating (DCR) quality degradation relationship.*

When signals are of good quality, ACR methods are insensible to little quality changes. In these cases, DCR methods are used, where two signals are presented to participants, who must assign a degradation quality value of each other according to Table 2.5. The mean value assigned by participants is DMOS.

Normally, original signal is presented and then degraded signal (transmitted signal).

2.3.1.2 Objective methods

Intrusive methods usually use two input signals, one as a reference (original signal) and another one degraded (transmitted signal). They can be classified in two main groups: *Time Domain Methods* and *Perception Domain Methods.*

Signal to Noise Ratio (SNR) and Peak signal-to-noise ratio (PSNR) are two examples of Time Domain Methods. This methods are easy to implement, but the correlation with subjective measures is not good [41][3][42][26][6][38].

Perception Domain Methods make perception relevant measures, transforming signals to perception domain using human audio perception models. These methods are more complex but present better correlation with subjective methods. Typical Perception Domain Methods are Perceptual Speech Quality (PSQM), Measuring Normalizing Blocks (MNB), Enhanced Modified Bark Spectral Distortion (EMBSD) and Perceptual Evaluation of Speech

Quality (PESQ). Noteworthy that PESQ includes a alignment mechanism between signals and gives its quality score in MOS scale.

Non-intrusive methods do not need extra data injection to measure the performance and audio quality. These methods can be classified in parameter-based and signal-based methods. The latter's predict quality only using the degraded signal (NULL reference method). The former's predict quality from IP network parameters such as packet loss, jitter, delay and non specific network parameters as codec, eco, bit rate, etc. Examples of these methods are E-Model and PSQA.

E-Model is and empirical mathematic model standardized by ITU [23]. E-Model is a set of formulas that take public switched telephone network (PSTN) parameters and packet switching network parameters as input and returns the associated quality factor. Although is a model used for network planning, nowadays is very used to predict perceived quality in VoIP.

2.4 Cloud Computing

Cloud computing[11], often referred to as simply "the cloud", is the delivery of on-demand computing resources-everything from applications to data centres-over the Internet and on a pay-for-use basis. A cloud is a powerful combination of computing, networking, storage, management solutions, and business applications that facilitate a new generation of IT and consumer services. These services are available on demand and are delivered economically without compromising security or functionality[28][18].

Cloud computing relies on sharing of resources to achieve coherence and economies of scale similar to a utility (like the electricity grid) over a network. At the foundation of cloud

[11] http://www.ibm.com/cloud-computing/us/en/what-is-cloud-computing.html

computing is the broader concept of converged infrastructure and shared services.

The cloud also focuses on maximizing the effectiveness of the shared resources. Cloud resources are usually not only shared by multiple users but as well as dynamically re-allocated as per demand. This can work for allocating resources to users in different time zones. For example, a cloud computer facility which serves European users during European business hours with a specific application (eg. email) while the same resources are getting reallocated and serve North American users during North America's business hours with another application (eg. web server). This approach should maximize the use of computing powers thus reducing environmental damage as well, since less power, air conditioning, rackspace, and so on, is required for the same functions. The term moving cloud also refers to an organization moving away from a traditional capex model (Capital expenditures, buy the dedicated hardware and depreciate it over a period of time) to the opex model (Operational expenditure, use a shared cloud infrastructure and pay as you use it).

Proponents claim that cloud computing allows companies to avoid upfront infrastructure costs, and focus on projects that differentiate their businesses instead of infrastructure. Proponents also claim that cloud computing allows enterprises to get their applications up and running faster, with improved manageability and less maintenance, and enables IT to more rapidly adjust resources to meet fluctuating and unpredictable business demand.

Cloud Computing is the result of evolution and adoption of existing technologies and paradigms [Fig. 2.9]. The goal of cloud computing is to allow users to take advantage from all of these technologies, without the need for deep knowledge about or expertise with each one of them. The cloud aims to cut costs and help the users focus on their core business instead of being impeded by IT obstacles.

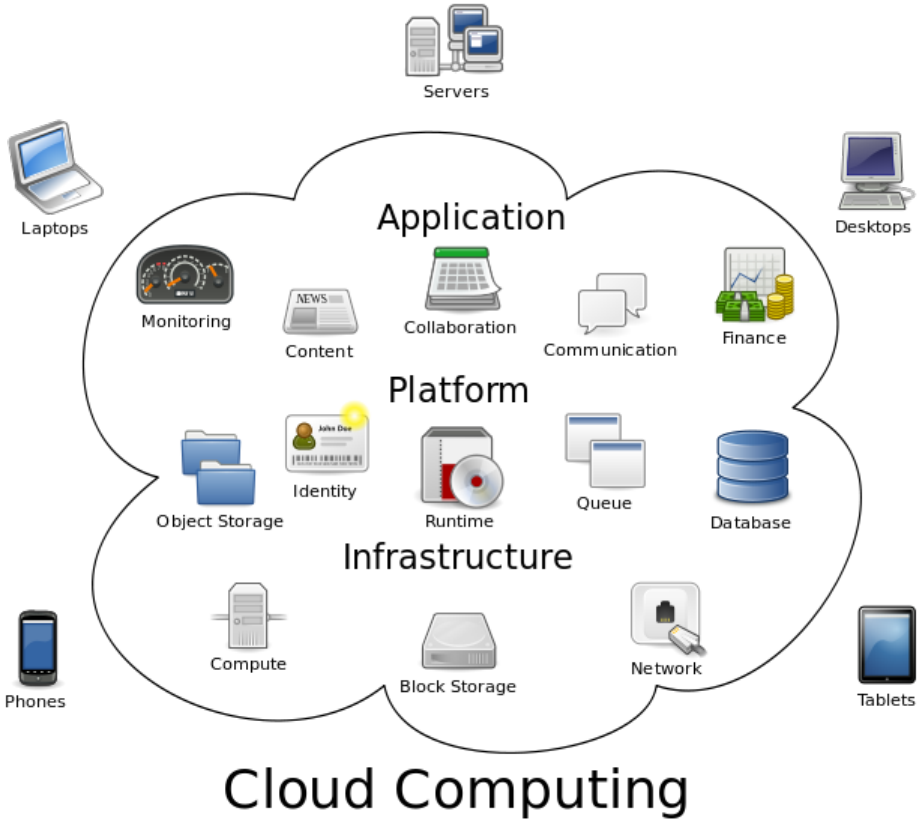

Figure 2.9: *Cloud computing logical diagram.*

The main enabling technology for cloud computing is virtualization. Virtualization abstracts the physical infrastructure, which is the most rigid component, and makes it available as a soft component that is easy to use and manage. By doing so, virtualization provides the agility required to speed up IT operations, and reduces cost by increasing infrastructure utilization. On the other hand, autonomic computing automates the process through which the user can provision resources on-demand. By minimizing user involvement, automation speeds up the process and reduces the possibility of human errors.

Cloud computing also leverages concepts from utility computing in order to provide metrics for the services used. Such metrics are at the core of the public cloud pay-per-use

models. In addition, measured services are an essential part of the feedback loop in autonomic computing, allowing services to scale on-demand and to perform automatic failure recovery.

Cloud computing is a kind of grid computing; it has evolved from grid computing by addressing the QoS (quality of service) and reliability problems. Cloud computing provides the tools and technologies to build data/compute intensive parallel applications with much more affordable prices compared to traditional parallel computing techniques.

The National Institute of Standards and Technology's definition of cloud computing identifies "five essential characteristics":

- **On-demand self-service**. A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.
- **Broad network access**. Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).
- **Resource pooling**. The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. ...
- **Rapid elasticity**. Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear unlimited and can be appropriated in any quantity at any time.
- **Measured service**. Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type

of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.

Cloud computing providers offer their services according to several fundamental models [Fig. 2.10]: infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS) where IaaS is the most basic and each higher model abstracts from the details of the lower models.

Infrastructure as a Service (IaaS) is the most basic cloud-service model, where providers of IaaS, offer computers (physical or more often virtual machines) and other resources. IaaS clouds often offer additional resources such as a virtual-machine disk image library, raw (block) and file-based storage, firewalls, load balancers, IP addresses, virtual local area networks (VLANs), and software bundles. IaaS-cloud providers supply these resources on-demand from their large pools installed in data centres. For wide-area connectivity, customers can use either the Internet or carrier clouds (dedicated virtual private networks).

To deploy their applications, cloud users install operating-system images and their application software on the cloud infrastructure. In this model, the cloud user patches and maintains the operating systems and the application software. Cloud providers typically bill IaaS services on a utility computing basis: cost reflects the amount of resources allocated and consumed. Typical IaaS providers include: Amazon EC2, AirVM, Azure Services Platform, DynDNS, Google Compute Engine, HP Cloud, iland, Joyent, LeaseWeb, Linode, NaviSite, Oracle Infrastructure as a Service, Rackspace, ReadySpace Cloud Services, ReliaCloud, SAVVIS, SingleHop, and Terremark.

In **Platform as a Service** (PaaS), cloud providers deliver a computing platform typically including operating system, programming language execution environment, database, and web server. Application developers can develop and run their software solutions on a cloud platform without the cost and complexity of buying and managing the underlying hardware and software layers. With some PaaS offers, the underlying computer and storage resources scale automatically to match application demand such that cloud user does not have to allocate resources manually. Examples of PaaS include: AWS Elastic Beanstalk, Cloud Foundry, Heroku, Force.com, EngineYard, Mendix, OpenShift, Google App Engine, AppScale, Windows Azure Cloud Services and OrangeScape.

Software as a Service (SaaS) is used in business model where users are provided access to application software and databases. Cloud providers manage the infrastructure and platforms that run the applications. SaaS is sometimes referred to as "on-demand software" and is usually priced on a pay-per-use basis. SaaS providers generally price applications using a subscription fee.

In the SaaS model, cloud providers install and operate application software in the cloud and users access the software from cloud clients. Cloud users do not manage the cloud infrastructure and platform where the application runs. This eliminates the need to install and run the application on the user's own computers, which simplifies maintenance and support. Cloud applications are different from other applications in their scalability (which can be achieved by cloning tasks onto multiple virtual machines at run-time to meet changing work demand). Load balancers distribute the work over the set of virtual machines. This process is transparent to the cloud user, who sees only a single access point. To accommodate a large number of cloud users, cloud applications can be multi-tenant, that is, any machine serves more than one cloud user organization. It is common to refer to special types of cloud based application software with a similar naming convention: desktop as a service,

business process as a service, test environment as a service, communication as a service. The pricing model for SaaS applications is typically a monthly or yearly flat fee per user, so price is scalable and adjustable if users are added or removed at any point. Examples of SaaS include: Google Apps, Microsoft Office 365, Petrosoft, Onlive, GT Nexus, Marketo, Casengo, TradeCard and CallidusCloud.

Proponents claim SaaS allows a business the potential to reduce IT operational costs by outsourcing hardware and software maintenance and support to the cloud provider. This enables the business to reallocate IT operations costs away from hardware/software spending and personnel expenses, towards meeting other goals. In addition, with applications hosted centrally, updates can be released without the need for users to install new software. One drawback of SaaS is that the users' data are stored on the cloud provider's server. As a result, there could be unauthorized access to the data.

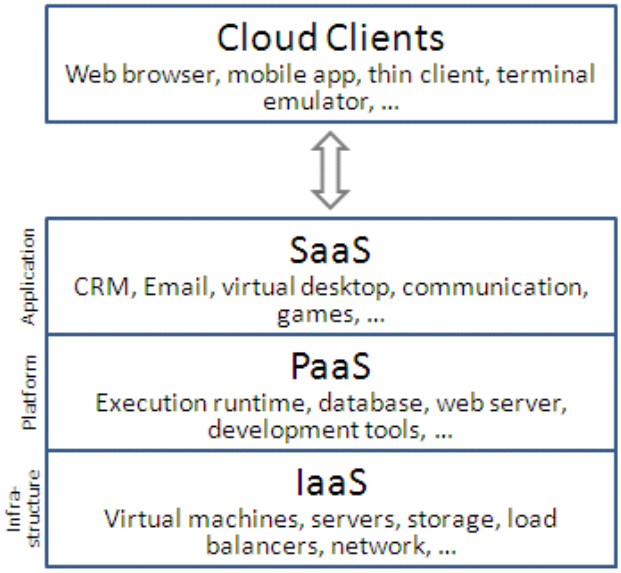

Figure 2.10: *Cloud computing layers.*

With most organizations focusing on leveraging the cloud in order to cut capital expenditure and control operating costs, there is aggressive growth in business for cloud adoption. However, the cloud can bring security risks and challenges for IT Management, which can be

more expensive for the organization to deal with, even considering the cost saving achieved by moving to the cloud. Therefore, it is very important for businesses to understand their requirements before opting for various deployment models available on the cloud.

There are primarily four cloud deployment models [Fig. 2.11], which are discussed below, along with scenarios in which a business could opt for each. These models have been recommended by the National Institute of Standards and Technology (NIST).

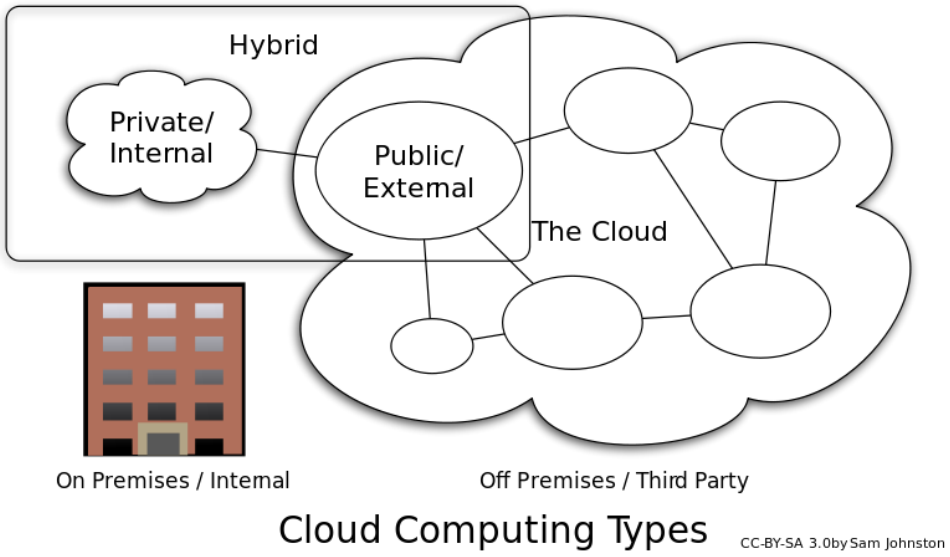

Figure 2.11: *Cloud computing deployment models.*

Private cloud is cloud infrastructure operated solely for a single organization, whether managed internally or by a third-party and hosted internally or externally. Undertaking a private cloud project requires a significant level and degree of engagement to virtualize the business environment, and requires the organization to reevaluate decisions about existing resources. This model can improve business, but every step in the project raises security issues that must be addressed to prevent serious vulnerabilities.

	Public Cloud	Private Cloud
Initial cost	Typically zero	Typically high
Running cost	Predictable	Unpredictable
Customization	Limited	Possible
Privacy	No	Yes
Single sign-on	Impossible	Possible
Scaling	Easy within defined limits	Laborious but no limits

Table 2.6: *Public and Private cloud models comparison for SaaS.*

In a **public cloud**, services are rendered over a network that is open for public use. Technically there is no difference between public and private cloud architecture, however, security consideration may be substantially different for services (applications, storage, and other resources) that are made available by a service provider for a public audience and when communication is effected over a non-trusted network. Generally, public cloud service providers like Amazon AWS, Microsoft and Google own and operate the infrastructure and offer access only via Internet.

Community cloud shares infrastructure between several organizations from a specific community with common concerns (security, compliance, jurisdiction, etc.), whether managed internally or by a third-party and hosted internally or externally. The costs are spread over fewer users than a public cloud (but more than a private cloud), so only some of the cost savings potential of cloud computing are realized.

Hybrid Cloud is a composition of two or more clouds (private, community or public) that remain unique entities but are bound together, offering the benefits of multiple deployment models. Such composition expands deployment options for cloud services, allowing IT organizations to use public cloud computing resources to meet temporary needs. This capability enables hybrid clouds to employ cloud bursting for scaling across clouds. Cloud bursting is an application deployment model in which an application runs in a private cloud

or data center and "bursts" to a public cloud when the demand for computing capacity increases. A primary advantage of cloud bursting and a hybrid cloud model is that an organization only pays for extra compute resources when they are needed. Therefore, cloud bursting enables data centres to create an in-house IT infrastructure that supports average workloads, and use cloud resources from public or private clouds, during spikes in processing demands. By utilizing an hybrid cloud architecture, companies and individuals are able to obtain degrees of fault tolerance combined with locally immediate usability without dependency on internet connectivity. Hybrid cloud architecture requires both on-premises resources and off-site (remote) server-based cloud infrastructure.

Fig. 2.12 show how each sample services would operate per development model:

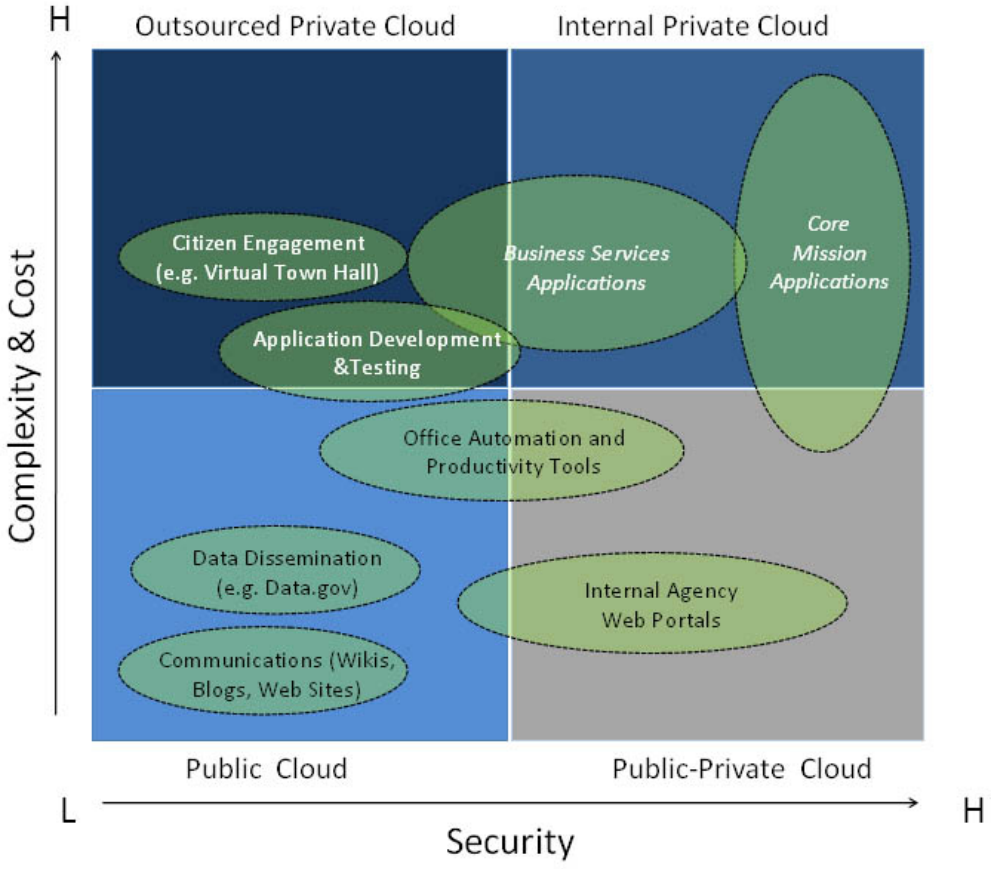

Figure 2.12: *Sample services operating in cloud models.*

Summarizing, cloud computing, independently from its deployment and service model, exhibits the following benefits:

- **Agility**: improves users' ability to re-provision technological infrastructure resources.
- **Cost reduction**: as infrastructure is typically provided by a third-party and does not need to be purchased for one-time or infrequent intensive computing tasks. Pricing on a utility computing basis is fine-grained with usage-based options and fewer IT skills are required for implementation (in-house).
- **Virtualization**: allows servers and storage devices to be shared and utilization be increased. Applications can be easily migrated from one physical server to another.
- **Device and location independence**: enable users to access systems using a web browser regardless of their location or what device they are using.
- **Multi-tenancy**: enables sharing of resources and costs across a large pool of users thus allowing for centralization of infrastructure (lower costs), peak-load capacity increases and use and efficiency improvements for systems that are often underutilized.
- **Reliability**: improved if multiple redundant sites are used, which makes well-designed cloud computing suitable for business continuity and disaster recovery.
- **Scalability and elasticity**: dynamic ("on-demand") provisioning of resources on a fine-grained, self-service basis near real-time.
- **Performance monitoring**
- Maintenance: easier because not need to be installed on each user's computer and can be accessed from different places.

On the other hand, security issues concern cloud users due to loss of control over certain sensitive data, and the lack of security for stored kernels. The complexity of security is

greatly increased when data is distributed over a wider area or greater number of devices and in multi-tenant systems that are being shared by unrelated users. In addition, user access to security audit logs may be difficult or impossible. Private cloud installations are in part motivated by users' desire to retain control over the infrastructure and avoid losing control of information security

2.5 Conclusion

Benefits and features of cloud computing like fast deployment, multi-tenancy, reliability, elasticity and specially cost reduction in given the current economic crisis, makes cloud attractive to the migration of companies services. Because of 56% of European decision-makers estimate that the cloud is a priority between 2013 and 2014 [12]. However, a cloud migration can present numerous challenges such as performance and raise security concerns.

Communication is the most popular use of the Internet. Although most of the technologies that are unique to the Internet communication are done in text, there is also Internet telephony. Voice over IP (VoIP) is a methodology and group of technologies for the delivery of voice communications and multimedia sessions over Internet Protocol (IP) networks, such as the Internet, being a direct solution to Internet telephony.

Many network parameters like latency, jitter or packet loss directly affect to VoIP communication quality. Moreover, user experience not only depends on network characteristics, but coding [Table 2.4], compression and recovery algorithms, determining that the quality of service for a user the quality of service will be for a user what he can perceived of it (PQoS), regardless of the network state what he can perceived of it (PQoS), regardless of the network state.

Perceived Quality of Service (PQoS) measures could be made using objective o subjective methods being subjective methods the best accepted metric since represent a direct connection with users perceived quality. Nevertheless, some objective methods have a good correlation with subjectives methods such as Perceptual Evaluation of Speech Quality (PESQ) and Perceptual Evaluation of Audio Quality (PEAQ) [31].

To cover the problem of security in the public cloud and communications, several technologies and methods coexist to prevent unauthorized interceptors from accessing telecommunications in an intelligible form, while still delivering content to the intended recipients. Encryption, Virtual Lan (VLAN) and Virtual Private Network (VPN) are security technologies to resolve the three main problems: Confidentiality, Integrity and Availability. VPN is based on tunnelling, ensuring the provision of confidentiality and integrity of voice packets in IP technology.Moreover, tunneling makes use of Authentication Protocol prior to establishing a connection. VPN technique consists of encapsulation. Voice traffic is secured by encapsulating it inside a tunneling standard, whose fundamental mechanism is encryption that ensures confidentiality and data integrity in VoIP.

In the next chapter, a secure teleconferencing model is presented to cover security issues in cloud computing and teleconference communication through a public cloud, taking into account the real-time drawbacks and speech quality in VoIP.

Chapter 3

Proposal

Security issues are the major concerns about public cloud computing and VoIP. Other issues arising from VoIP and similar real-time software are latency and its repercussion in the quality of service. Two new teleconferencing secure architectures are presented to overcome these issues.

Secure teleconferencing architecture models proposed are: **VPN Secure Cloud Teleconferencing** and **Hidden Server Secure Cloud Teleconferencing**. Moreover, a baseline model is presented to compare results. The former consists in establishing two VPN connections to the VoIP teleconference server during a call to ensure confidentiality and integrity through data encryption [see 4.2]. The latter involves two extra nodes that will act as firewalls, filtering and forwarding VoIP traffic to the teleconference server [see 4.3]. This architecture hides Asterisk server, preventing it from DoS and VoIP component attacks [2.2].

To analyse viability, security and real-time traffic issues, *latency* and *speech quality* are measured. Speech quality will be estimated with PQoS time domain intrusive objective methods SNR and PSNR, a perception domain intrusive objective method PESQ and the MOS subjective method. Correlation between objective methods and MOS will be studied to determine a speech quality estimator for future work, without necessity of applying MOS method.

Instance Name	Virtual Cores / Compute Units	RAM Memory	Storage (GB)	Platform	I/O performance	API Name
Micro	1/up to 2 in short bursts	615 MiB	None (use Amazon EBS volumes for storage)	32/64 bits	Very Low	t1.micro
Small	1/1	1.7 GiB	160 GB	32/64 bits	Low	m1.small
Large	2/4	7.5 GiB	2 x 420 GB	64 bits	Moderate	m1.large

Table 3.1: *Amazon Web Services EC2 Instances used for Secure Cloud Teleconferencing models proposed.*

Amazon Web Services EC2 (AWS EC2) will be used as the cloud computing platform. AWS EC2 is a flexible platform with multiple configuration options like operating systems, type of instances and firewall and networking. Secure Architectures proposed will rely over three different type of instances: Micro Instances, Small Instances and Large Instances [Table 3.1].

Asterisk is chosen as communication application and IP PBX system and will be installed in AWS EC2 instance server. Asterisk is free and open source and supports a wide range of Voice over IP protocols. **SIP protocol** is configured for VoIP calls. Audio Codec behaviour will be also analysed, configuring Asterisk server and peers to support four different open source audio codecs to study bit rate and compression delay influence in Speech Quality: GSM, G.711 a-law, G.711 μ-law and G.726-32 [Table 3.2].

Audio Codec	Sample Rate	Bit rate	Latency
GSM	8 KHz	13 kbps	20-30 ms
G.711 a-law	8 KHz	64 kbps	0.125 ms
G.711 μ-law	8 KHz	64 kbps	0.125 ms
G.726-32	8 KHz	32 Kbps	0.125 ms

Table 3.2: *Audio Codecs used for Asterisk VoIP PBX.*

All peers are **Android**[1] **4.0+ devices** and use **CSipSimple**[2] **client** to connect to the Asterisk server through WiFi networks in their LANs. For VPN connections, peers are configured with **FEAT VPN**[3] to use 1024 bits public key authentication and SHA-1 algorithm for encryption.

In the next chapter, VPN Secure Cloud Teleconferencing and Hidden Server Cloud Teleconferencing architectures and configuration are presented as well as the baseline model.

[1]Android is a Linux-based operating system designed primarily for touchscreen mobile devices such as smartphones and tablet computers. Android is open source and Google releases the code under the Apache License.

[2]CSipSimple is a Voice over Internet Protocol (VoIP) application for Google Android operating system using the Session Initiation Protocol (SIP). It is open source and free software released under the GNU General Public License.

[3]FEAT VPN is an Android application that brings OpenVPN to Android versions 2.1 through 4.2. In contrast to existing applications, FEAT VPN does not require you to root your Android phone or tablet. FEAT VPN works on unmodified off-the-shelf devices.

Part II

Experiments and results

Chapter 4

System Architecture

VPN Secure Cloud Teleconferencing and Hidden Server Secure Cloud Teleconferencing architecture proposed models and the baseline architecture (Basic Architecture) are set in *AWS EC2 public cloud* and launched in *US-east-1d zone*. All instances run a *64bit 12.10 Ubuntu*[1] operating system. Some considerations need to be taken about the deployed Asterisk server in all architectures.

As *Asterisk* has been chosen as IP PBX and communication platform, it has to be installed and configured in the server. *SIP protocol* is configured through sip.conf file as follows:

- *UDP protocol* is enabled.
- *canreinvite/directmedia: off*. Every packet passes through the Asterisk server, not allowing direct connection once communication is established.
- *qualify: yes*. Enabled to do not let firewalls and routers delete routes.
- *alwaysauthreject: yes*. Reject bad authentication requests on valid usernames with the

[1]Ubuntu is a computer operating system based on the Debian Linux distribution and distributed as free and open source software, using its own desktop environment. As of 2012, according to online surveys, Ubuntu is the most popular Linux-based operating system on desktop/laptop personal computers, and most Ubuntu coverage focuses on its use in that market. However, it is also popular on servers and for cloud computing.

same rejection information as with invalid usernames, denying remote attackers the ability to detect existing extensions with brute-force guessing attacks.

- *tos (terms of service): 0x18.* Low delay and high throughput are set.
- *NAT* is enabled for peers but set to 'route' for the Asterisk server.
- *Peers' password must be strong*[2][7][19][13][35].
- *SIP port* is set to UDP *5060.*
- Allowed Audio Codecs are *GSM*, *G.711 a-law*, *G.711 μ-law* and *G.726-32* [Table 3.2].

Extra security measures could be set only accepting calls from determined IPs and denying all the rest, disabling ICMP echo-request, limiting established connections and new connections per minute to avoid DoS attacks and only opening both in the Asterisk server and in the AWS EC2 firewall ports: 5060 UDP (SIP protocol), 10000-20000 UDP (RTP protocol) and 1194 TCP (OpenVPN protocol, just for VPN Secure Cloud Teleconferencing Architecture).

Public cloud platform (AWS EC2) is considered secure, not the case of the rest of the network (Internet and LANs), considered an unsafe environment and attack susceptible.

[2]Strong passwords guidelines:

- Minimum password length of 12 to 14 characters.
- Generating passwords randomly where feasible.
- Avoiding passwords based on repetition, dictionary words, letter or number sequences, usernames, relative or pet names, romantic links (current or past), or biographical information (e.g., ID numbers, ancestors' names or dates).
- Including numbers, and symbols in passwords if allowed by the system.
- If the system recognizes case as significant, using capital and lower-case letters.
- Avoiding using the same password for multiple sites or purposes.
- Avoiding using something that the public or workmates know you strongly like or dislike.

CSipSimple (SIP VoIP client for Android OS) is installed and configured in all peers (Android 4.0+ devices). Audio Codecs allowed are *GSM*, *G.711 a-law*, *G.711 μ-law* and *G.726-32* and voice calls are automatically recorded. Each single VoIP client is previously signed up in the Asterisk server. Moreover, *echo cancellation* is enabled to avoid interferences and bad results.

In the next sections, proposed architectures are widely described as well as their particular configuration options if needed.

4.1 Basic Cloud Teleconferencing Architecture

The baseline cloud teleconference architecture consists in an Asterisk VoIP server launched in the AWS EC2 public cloud (specifically in US-east-1d zone). As mentioned in Chapter 4, every peer has previously been signed up in the Asterisk server, being able to call to others registered peers. For the further experiments, the architecture setup is shown in Fig. 4.1.

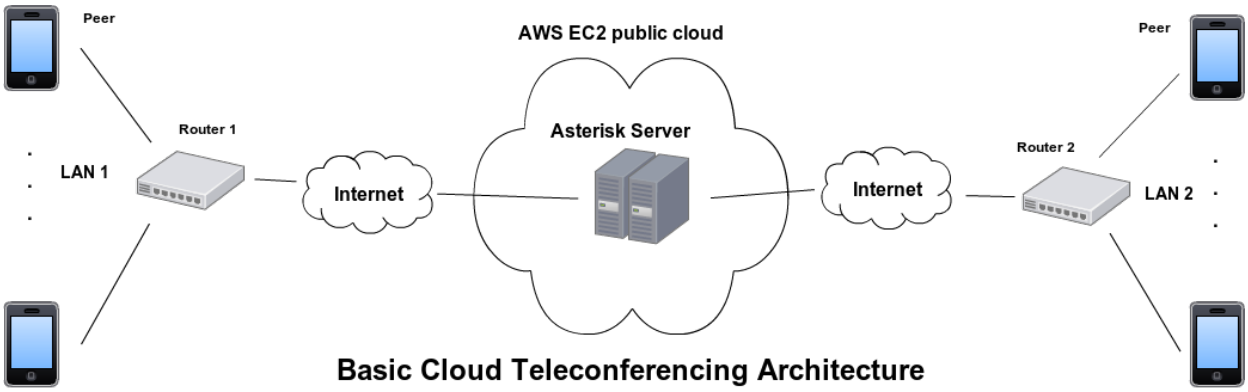

Figure 4.1: *Basic Cloud Teleconferencing Architecture.*

In Basic Cloud Teleconferencing Architecture [Fig. 4.1], no encryption or hidden server are set. Every peer can establish a VoIP communication to any of the rest of the peers without restriction. No extra parameters need to be configured both in the Asterisk server

and peers. This Basic Architecture will be a baseline to compare results with the proposed secure cloud architectures for VoIP teleconference.

4.2 VPN Secure Cloud Teleconferencing Architecture

The first proposal that addresses security issues in public clouds and communication is VPN Secure Cloud Teleconferencing Architecture. As the Basic Architecture, Asterisk VoIP server is launched in the AWS EC2 public cloud and peers are previously signed up.

Furthermore, *OpenVPN* and *OpenSSL* (*SSL/TLS public key authentication*) must be installed on the Asterisk server to provide VPN tunneling. An SSL session is established with bidirectional authentication between peers before a call (i.e. each side of the connection must present its own certificate). If the SSL/TLS authentication succeeds, encryption/decryption and HMAC key source material is then randomly generated by OpenSSL's RAND_bytes function and exchanged over the SSL/TLS connection. Both sides of the connection contribute random source material. This mode never uses any key bidirectionally, so each peer has a distinct send HMAC, receive HMAC, packet encrypt, and packet decrypt key. Authentication certificates are generated of 1024 bits and *SHA-1 encryption algorithm* is used for voice data encryption. Authentication certificates are installed in peers once generated in the Asterisk-OpenVPN server and before establishing an VPN connection and a VoIP call.

The VPN Secure Cloud Teleconferencing Architecture setup is illustrated in Fig. 4.2, where each peer establish it's own VPN connection to the server before calling. VPN connections will be closed when VoIP call ends.

VPN Secure Cloud Teleconferencing Architecture resolves Authentication, Integrity and Confidentiality issues related to the security concept of VoIP [Table 2.5]. VPN Secure Cloud Teleconferencing Architecture prevent VoIP calls from attacks [see 2.2] such as:

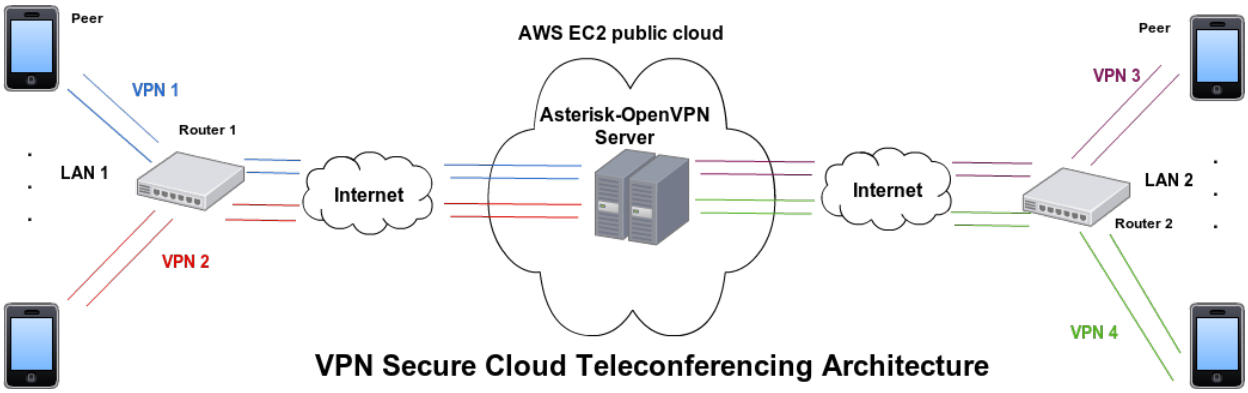

Figure 4.2: *VPN Secure Cloud Teleconferencing Architecture.*

- Registration attacks: covered by VPN public key based authentication.
- On call attacks: eavesdropping, sniffing, man-in-the-middle are solved by VPN encryption.
- Denial of Service attacks: these attacks are prevented by limiting the number of connections per minute and the total connections, denying no friend IPs, etc. [see 4].

However this architecture does not prevent from attacks on VoIP components such as the Asterisk server. Asterisk/OpenVPN server IP address is needed to establish a VoIP call, being know from all peer clients.

FEAT VPN is installed on Android peers to provide VPN connection with the server. Certificates and configuration files have been previously moved and installed on all clients.

Due to the VPN protocol (encryption/decryption and control flow data) delay issues may occur, affecting the perceived quality of service (PQoS). This repercussion needs to be studied to verify the VPN extra data load viability.

4.3 Hidden Server Secure Cloud Teleconferencing Architecture

The second proposed architecture is Hidden Server Secure Cloud Teleconferencing Architecture, establishing the Asterisk server in the AWS EC2 public cloud (US-east-1d zone). Two new instances (*End Points*) need to be launched in the same zone to complete the architecture. These instances will be always micro instances to reduce costs and will act as firewalls, hiding the server from the peers (VoIP clients) and forwarding VoIP traffic. Note that as in the other architectures, peers are previously signed up in the Asterisk server.

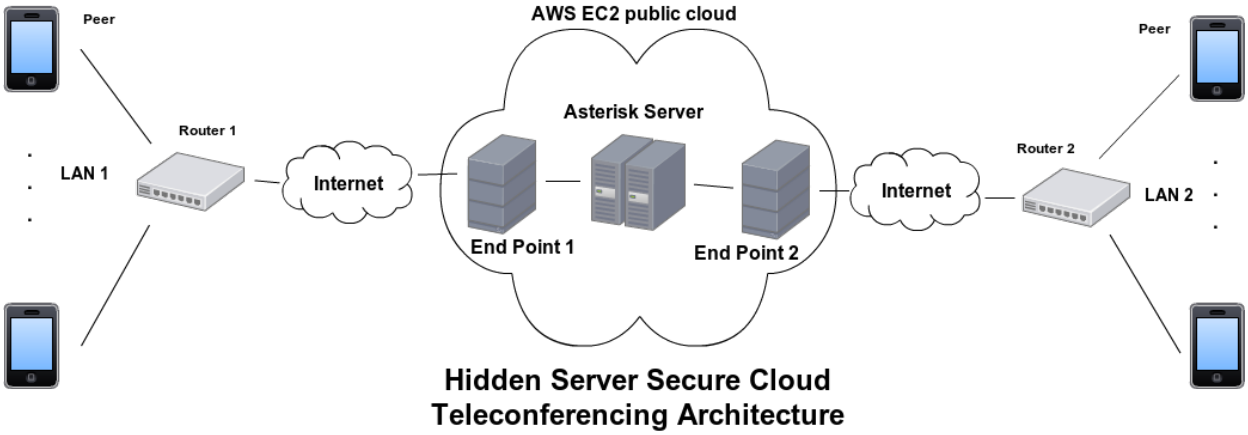

Figure 4.3: *Hidden Server Secure Cloud Teleconferencing Architecture.*

The Hidden Server Secure Cloud Teleconferencing Architecture setup is presented in Fig. 4.3. Socat is configured to create a new thread in the End Point per each new call and thus improve the performance of the End Point.

As End Points act like firewalls, *socat*[3] is installed in them to forward SIP control flow (port 5060) and RTP packets that contain voice traffic. End Points only forward pack-

[3]Socat is a command line based utility that establishes two bidirectional byte streams and transfers data between them. Because the streams can be constructed from a large set of different types of data sinks and sources, and because lots of address options may be applied to the streams, socat can be used for many different purposes.

ets from peers, denying packets from other sources. Establishing the End Points between peers and the Asterisk server, attacks on VoIP server are prevented due to the server IP address ignorance. Peers perform its SIP authentication and VoIP communication against End Points and not against the Asterisk server, only needing to know the server IP Address.

Although attacks on VoIP components are prevented in this architecture (as well as DoS attacks, see 4), there is neither data encryption nor secure authentication. Attacks that were cover in VPN Secure Cloud Teleconferencing Architecture such as registration attacks and on call attacks are not addressed.

In the following chapters, the proposed architectures are tested, extracting results to establish conclusions about security, performance and potential future work.

Chapter 5

Experiments

To test proposed architectures and could take conclusions about them in terms of security, costs and performance, a test plan must be define. Moreover, a new ratio must be defined to evaluate cost against the performance received.

5.1 Infrastructure

Proposed architectures and their particular configuration [see Chapter 4] have been deployed for its analysis. In each architecture, needed instances have been launched and configurated according to their particular security conditions.

Besides proposed architectures, other network parameters that affect the PQoS, need to be fixed before testing. Although Internet network conditions are unpredictable and can not

	Upload speed (kbps)	**Download speed (kbps)**	**Delay (ms)**
Day 1	241	2561	153
Day 2	264	2543	90
Day 3	260	2575	130
Day 4	265	2591	137
Day 5	265	2724	132
Average	259	2598.8	126.4

Table 5.1: *Experimental WiFi LAN Network Conditions.*

be establish, WiFi local network conditions from where peers connect to Asterisk server, could be determined. In Table 5.1, real WiFi LAN network conditions where measured during the testing week, taking a sample each day before testing. Tests are performed over 24 Mbps WiFi local networks to which peers are connected.

Initial tests yielded issues. Android mobile devices with lower performance than a 400 MHz processor do not computationally support FEAT VPN and CSipSimple execution at the same time. To prevent this initial issue, all peers will have at least the following specifications and applications installed:

- 1 GHz processor
- 1 GB RAM
- 50 MB of free storage
- Android 4.0+ OS
- FEAT VPN release 38
- CSipSimple 1.00.00 release 2225
- CSipSimple codec pack version 1.2 (for G.726 audio codec)

In the next section, test methodology is explained once the infrastructure is deployed and previous configurations are set.

5.2 Methodology

Once the infrastructures of the two proposed architectures are set, peers connect to the architecture to start testing. Depending on which architecture is deployed, peers connect in one way or another. If Basic Cloud Teleconferencing Architecture is deployed, peers use

CSipSimple to connect to the Asterisk server (through the server public IP address), registering with its caller id and its password. However, if VPN Secure Cloud Teleconferencing Architecture is deployed, peers first establish a VPN connection using FEAT VPN application and the public key certificates (previously provided) to connect to the VoIP server. Once connected through the VPN, peers register in the VoIP server via the private address provided in the established VPN connection, using CSipSimple. On the other hand, if Hidden Server Secure Cloud Teleconferencing Architecture is deployed, peers are registered in the End Points through the End Points public IP address and ignoring Asterisk server IP address.

For each kind of architecture, Asterisk server (Asterisk + OpenVPN server in VPN Secure Cloud Teleconferencing Architecture) is instanced with the three AWS instance types chosen: micro, small and large [Table 3.1]. Moreover, for each kind of Asterisk server instance, tests are made using the four audio codecs chosen: GSM, G.711 a-law, G.711 μ-law and G.726-32 [Table 3.2]. Architectures are tested using 2, 4, 6 and 8 peers to study quality degradation. A test experiment consist of the following steps:

- Depending on the number of peers being tested, peers register and all except two, establish a call in pairs from one LAN to the other with the current audio codec. The two remaining peers (one of each LAN) enable auto call recording and are placed in locked rooms without environmental noise.
- One of these two peers made a 10 seconds long call to the other one, divided in this sections:
  - First 1.5 seconds: silence.
  - Next 7 seconds say the following sentence in Spanish: "Estoy probando el proyecto de fin de máster. Estoy llamando a la extensión doscientos dos."
  - Last 1.5 seconds: silence.

- As auto call recording is enabled, emitted and received audio are recorded in WAV files to future analysis.

Once experiments are taken, emitted and received audio WAV files of each situation need to be processed. To extract SNR and PSNR values of each pair of emitted and received audio piece, WAVdiff[1] program is used. Furthermore, for speech quality analysis, PESQ algorithm is used. PESQ algorithm has been download from ITU webpage[21] and a wrapper[45] has been used to allow Matlab access to PESQ ITU's algorithm. This PESQ wrapper provides PESQ and PESQ-LQO values, being chosen the latter to easily compare with MOS scale. Also four people where asked to rank every received audio file in MOS scale, taking as result the average of the four scores. These people were asked to rank each audio received file in a closed room without any environmental noise.

In the next chapter, collected results are shown along with brief descriptions and comments of each.

[1]The WD (WavDiff) is a command-line tool for audio file comparison and statistical analysis. This tool accepts wav or raw PCM formats, provides fast automatic alignment, can compare files with different precision, for example IEEE double with 16-bit PCM and saves difference or aligned file.

Chapter 6

Results

Proposed architectures were tested by the methodology and the infrastructure described in the previous section. As MOS, PESQ, PSNR and SNR where measured per each pair of emitted and received audio, correlation between these metrics and MOS where analysed to see if any of them would be a good speech quality estimator.

SNR and PSNR have a extremely low correlation with MOS scores as shown in Table 6.1. SNR and PSNR would not be good speech quality estimators with these results. However, analysing PESQ and MOS correlation, PESQ has a correlation ratio of nearly 0.78, being an acceptable estimator for speech quality. As we have MOS scores per each situation and as it is the best metric to evaluate and determine speech quality, it is set to analyse the impact of security and network conditions in speech quality.

Delay effect in speech quality (MOS) has been also studied, being analysing in terms of

	Correlation
SNR-MOS	0.07
PSNR-MOS	0.06
PESQ-MOS	0.78

Table 6.1: *SNR, PSNR and PESQ MOS correlation.*

type of architecture, kind of instance and number of peers. As we can see in Fig. 6.1, delay does not affect in a uniform way to speech quality (a higher delay does not imply lower quality), although more low MOS scores are given with high delay. Moreover, delay impact is higher in VPN Secure Cloud Teleconferencing Architecture.

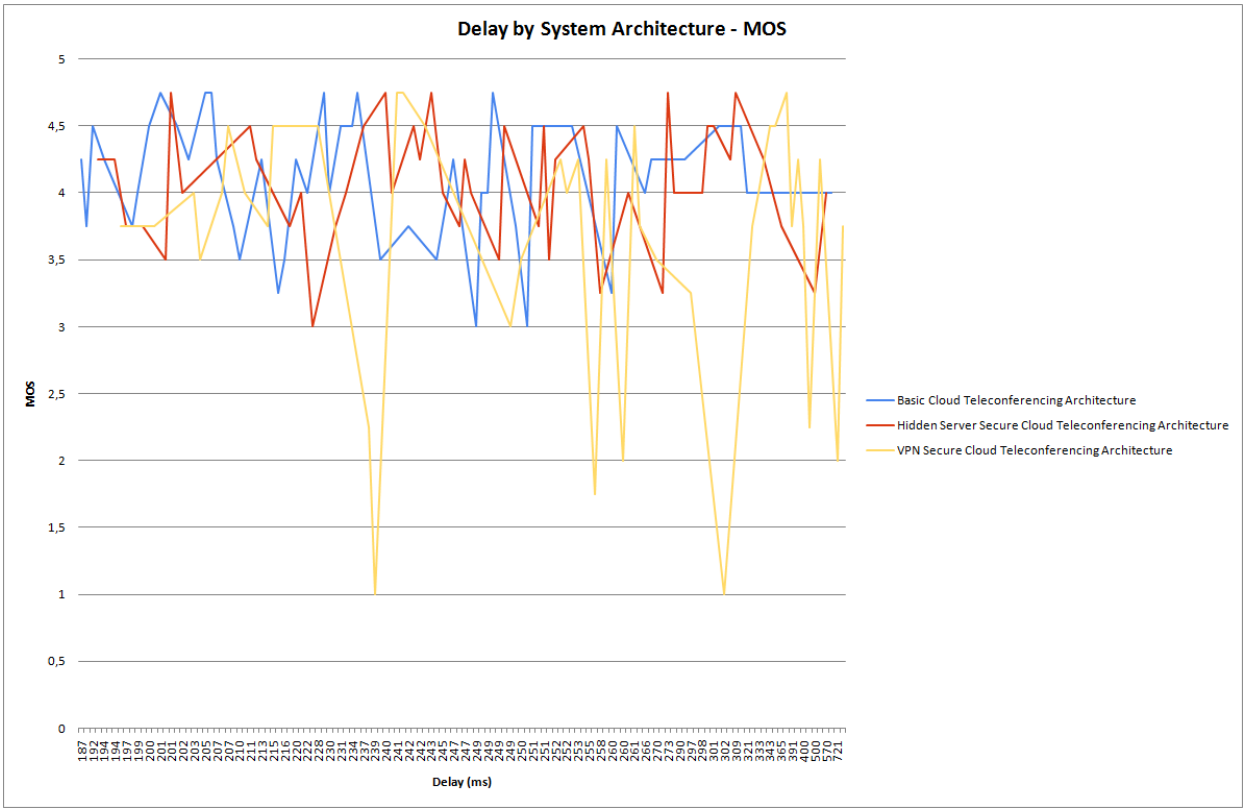

Figure 6.1: *Delay impact per cloud teleconferencing architecture.*

Similar results have been obtained in terms of kind of instance [Fig. 6.2]. Large instances are more influenced by delay that the Micro and Small instances.

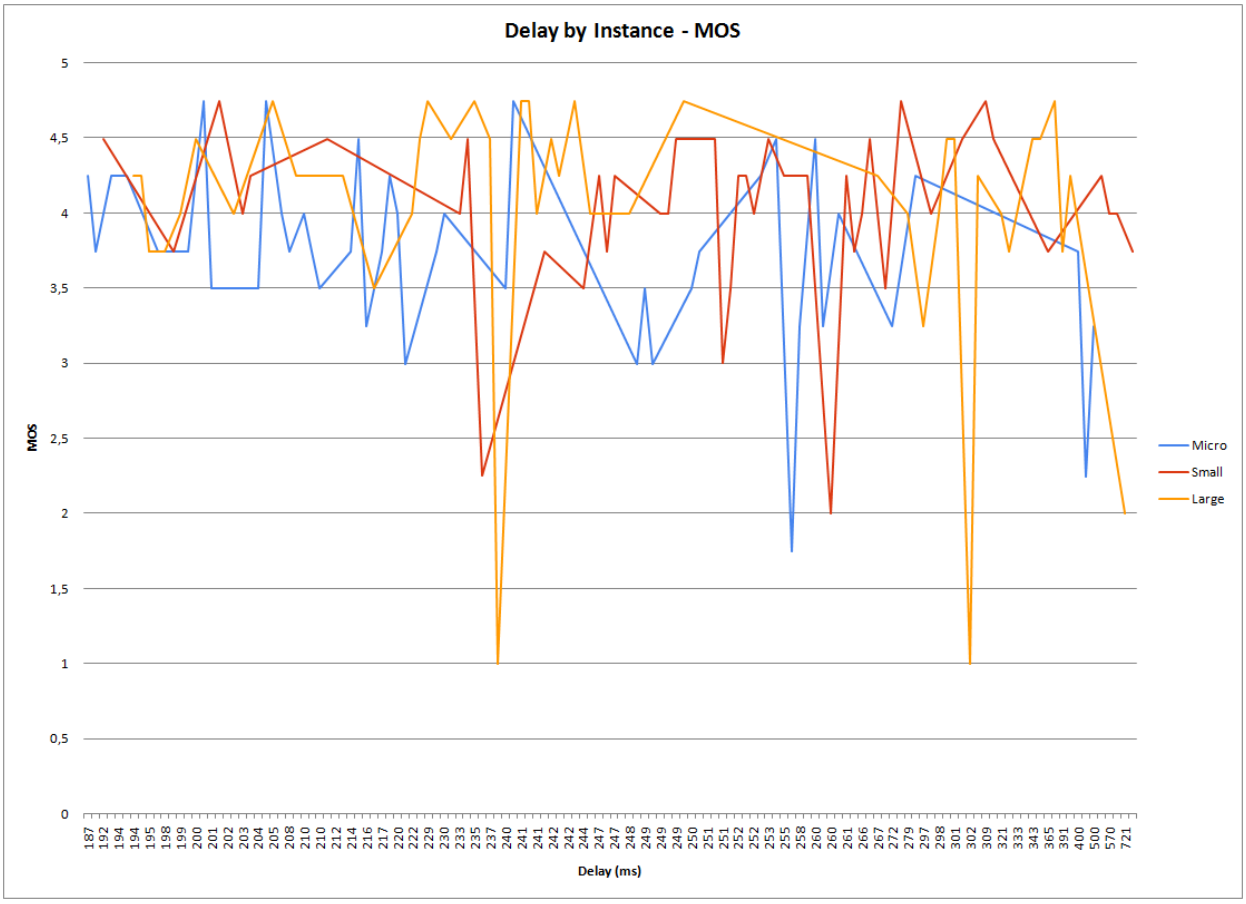

Figure 6.2: *Delay impact per kind of Amazon Web Services Instance.*

Regarding the number of peers, latency impact increases with the number of peers as shown in Fig. 6.3. The more number of peers, the more latency affects speech quality.

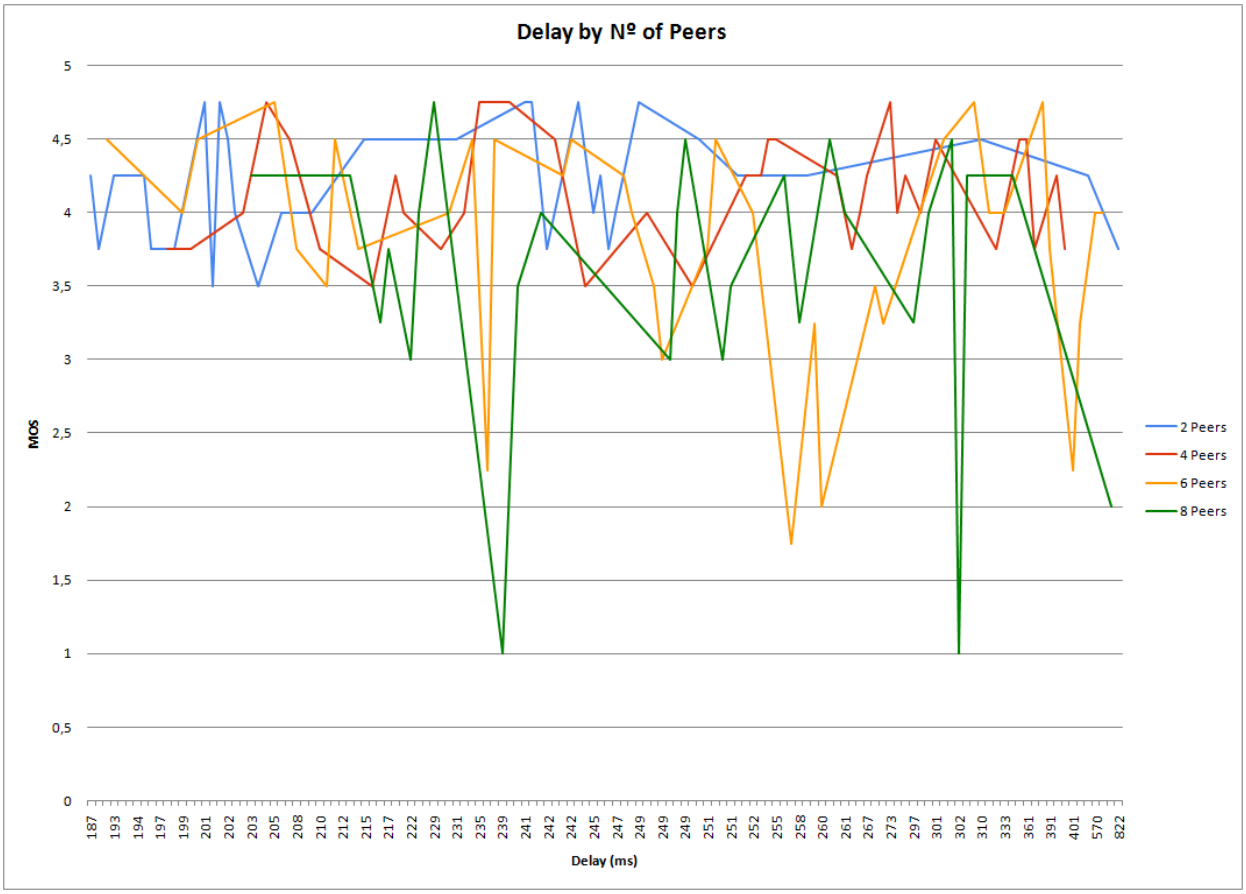

Figure 6.3: *Delay impact per number of peers.*

In the next subsections, speech quality collected results are presented by number of peers, architecture and audio codec. Furthermore, a new metric is fixed to study the architecture deployment costs and the speech quality obtained.

6.1 Speech Quality Results

Speech Quality values are presented by the number of peers for better comprehension. MOS quality values are shown per each proposed architecture according each audio codec.

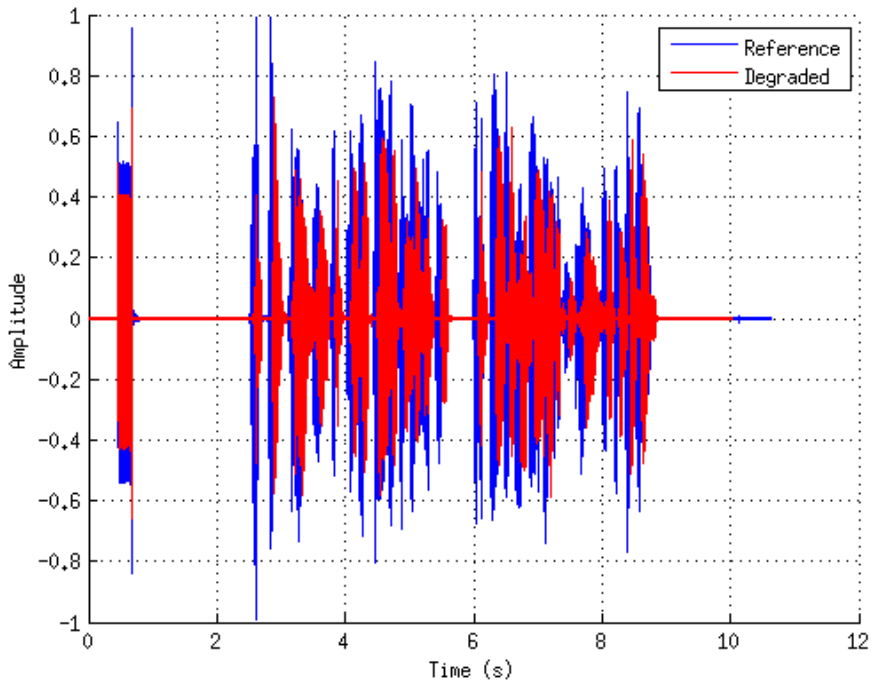

Figure 6.4: *Reference and degraded signal in an VPN Secure Cloud Architecture in a Large instance with 2 peers using G.711 μ-law audio codec and getting a MOS score of 4.75*

In Fig. 6.4 and Fig. 6.5 reference and degraded signal with an excellent and a very bad MOS score are presented respectively. Signal amplitude in the degraded signal with excellent MOS score [Fig. 6.4] almost overlaps reference signal amplitude, being significantly lower the degraded signal with low MOS score.

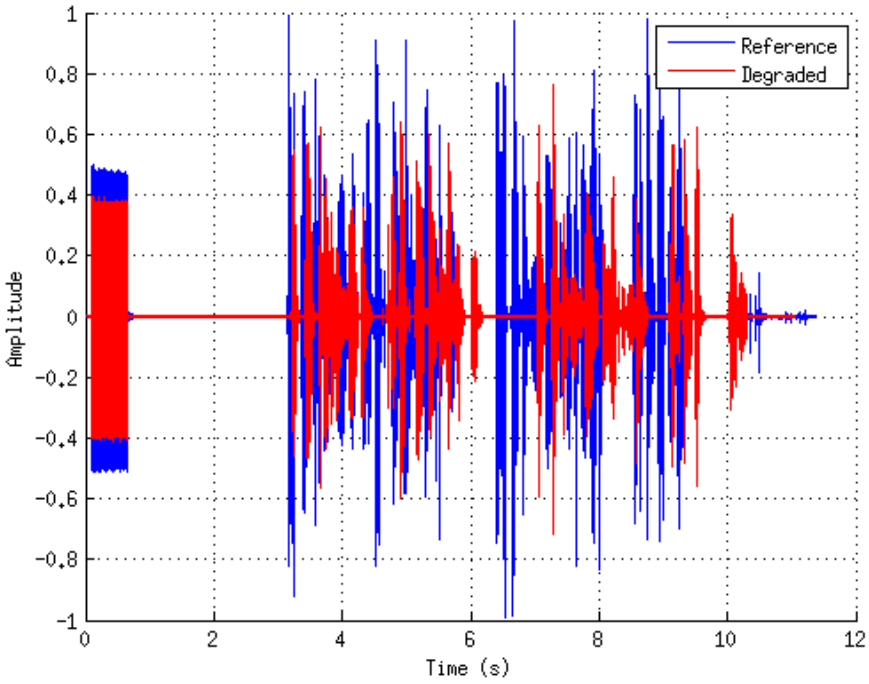

Figure 6.5: *Reference and degraded signal in an VPN Secure Cloud Architecture in a Small instance with 6 peers using G.711 μ-law audio codec and getting a MOS score of 2.*

6.1.1 2 peers Speech Quality

According to the results presented in Fig. 6.6, an instance with better performance does not represent a PQoS substantial improvement in Basic Cloud Teleconferencing Architecture.

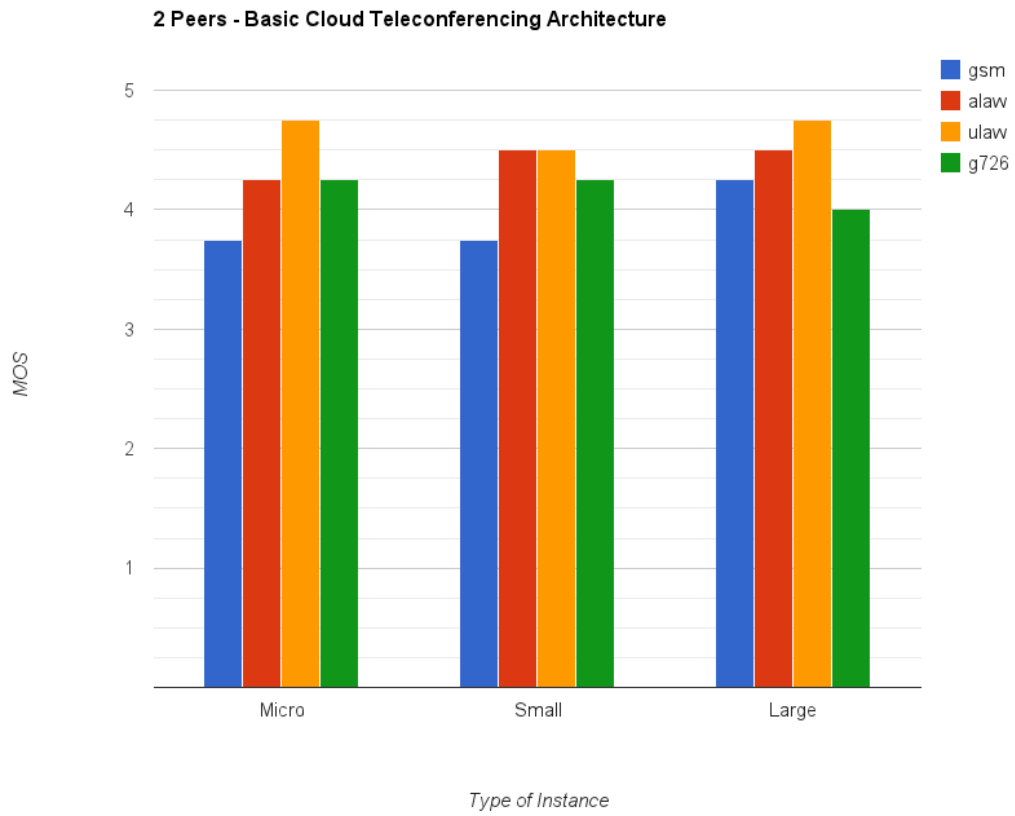

Figure 6.6: *Speech Quality in Basic Cloud Teleconferencing Architecture with 2 peers.*

In contrast, both VPN and Hidden Server Secure Cloud Teleconferencing Architectures, Micro instances provide slightly less MOS than Small and Large Instances as we can see in Fig. 6.7 and Fig. 6.8. The audio codec that provides better speech quality in most cases is G.711 μ-law.

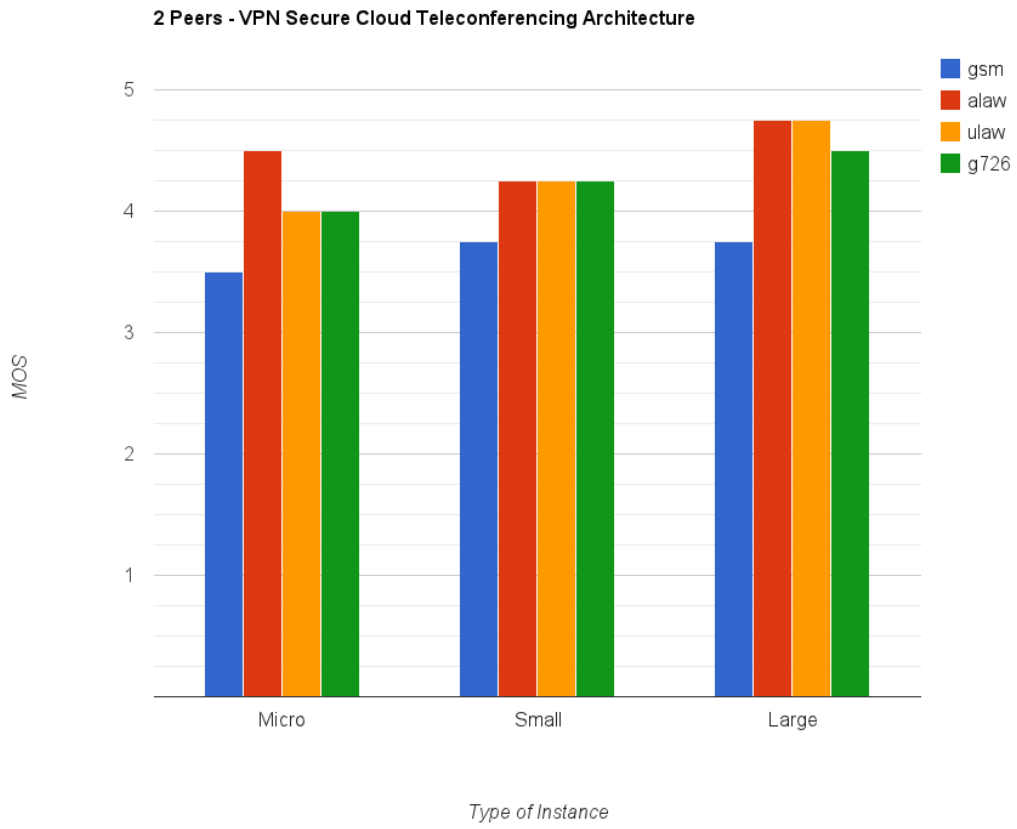

Figure 6.7: *Speech Quality in VPN Secure Cloud Teleconferencing Architecture with 2 peers.*

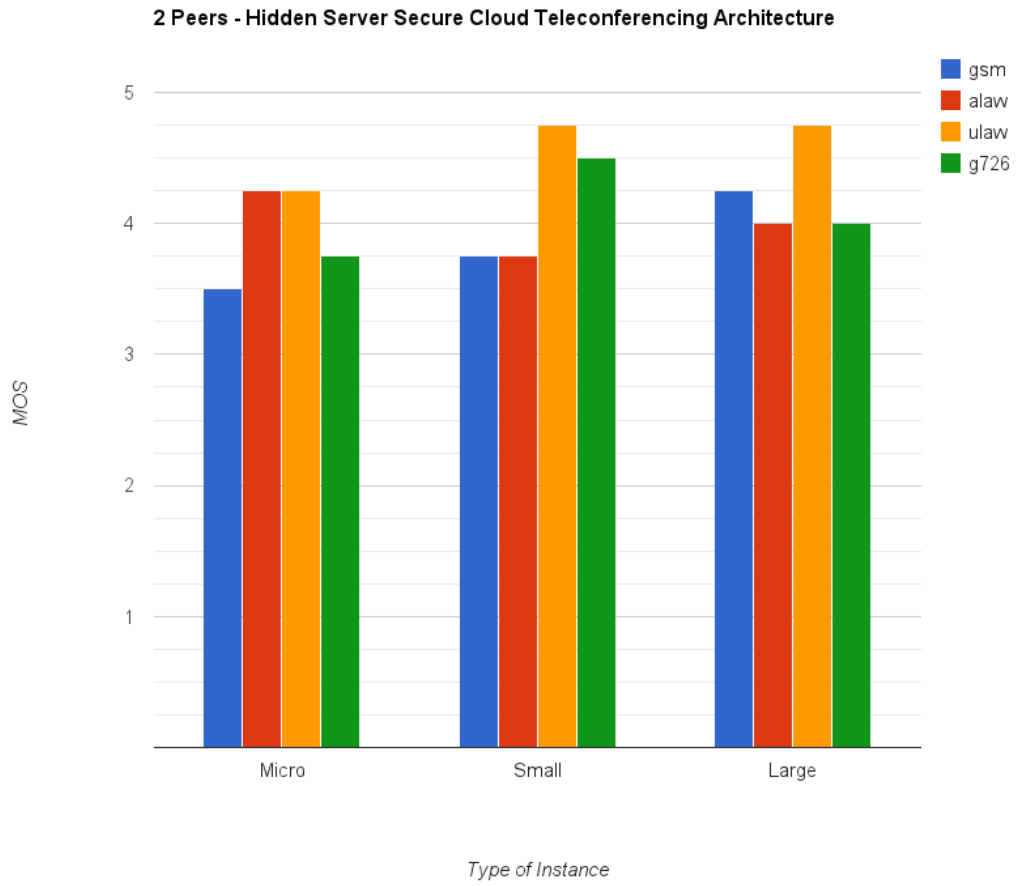

Figure 6.8: *Speech Quality in Hidden Server Secure Cloud Teleconferencing Architecture with 2 peers.*

6.1.2 4 peers Speech Quality

Basic Cloud Teleconferencing Architecture provides slightly less MOS in general [Fig. 6.9] than the proposed Architectures in spite of the extra load security when connecting 4 peers to the architectures. GSM audio codec gives less speech quality than G.711 a-law or G.711 μ-law in all cases, being the latters the audio codecs that provides the best speech quality in most cases.

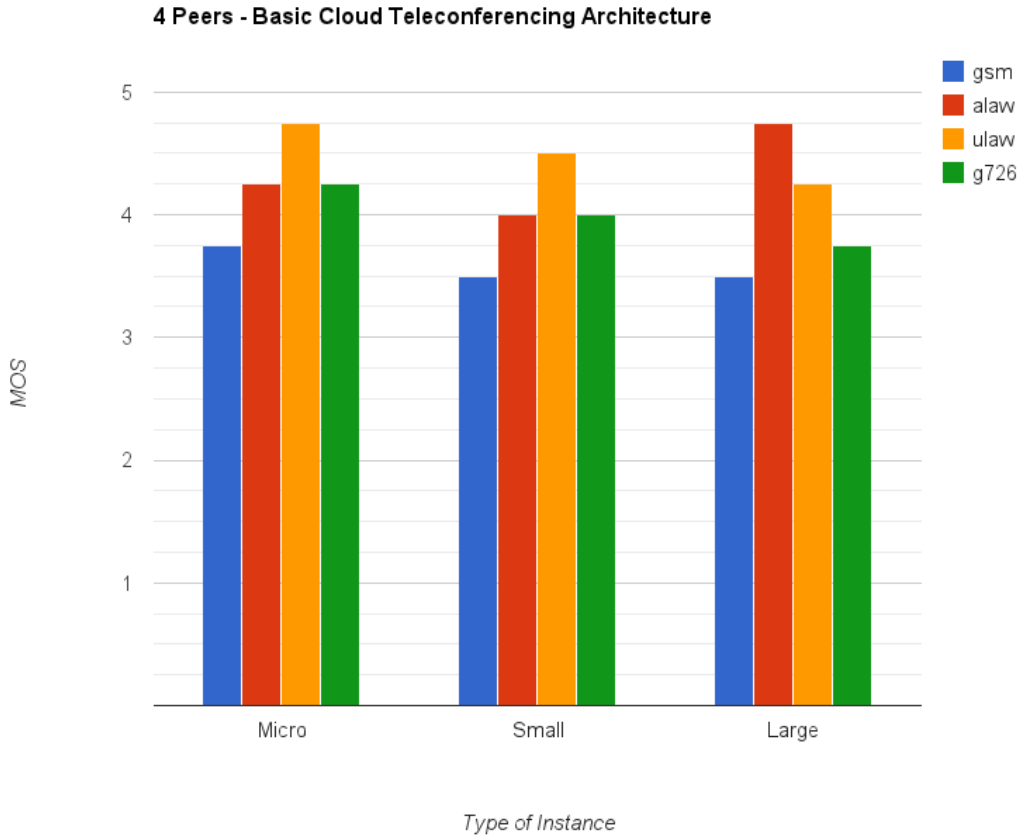

Figure 6.9: *Speech Quality in Basic Cloud Teleconferencing Architecture with 4 peers.*

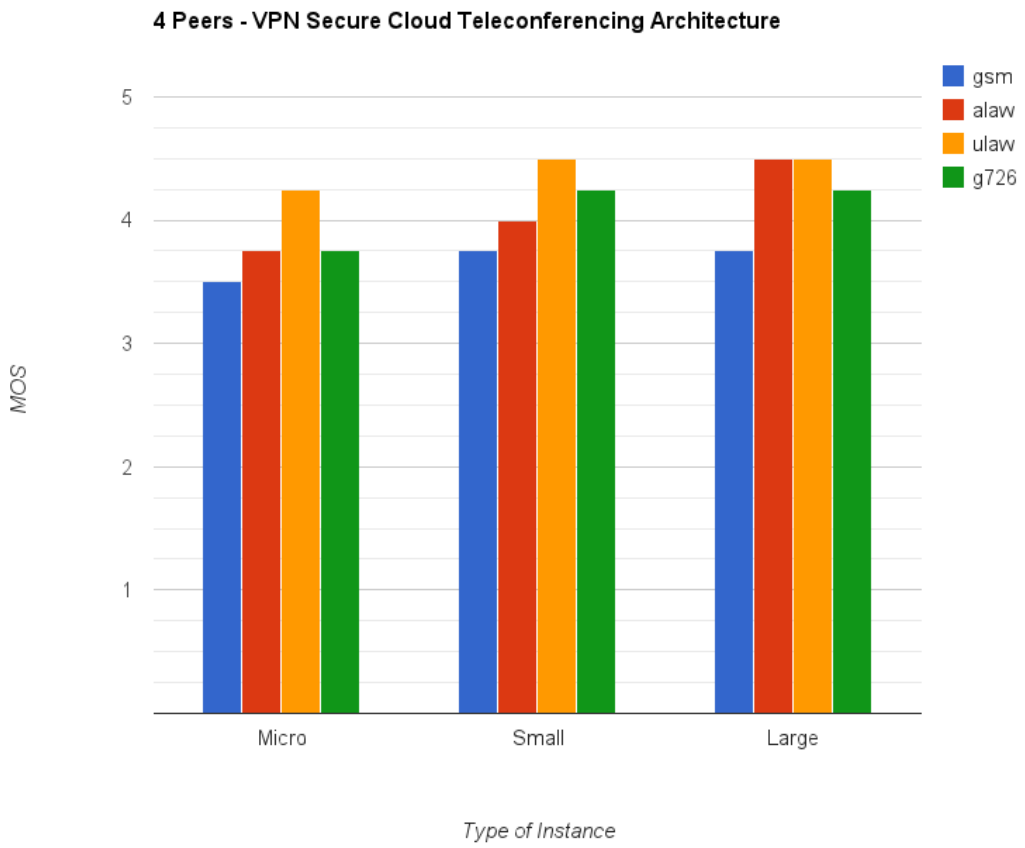

Figure 6.10: *Speech Quality in VPN Secure Cloud Teleconferencing Architecture with 4 peers.*

Using Micro instances, speech quality degrades more in VPN Architecture than in Basic or Hidden Server Architecture [Fig. 6.10]. When instance's computing capacity is enhanced, this degradation disappears. Unlike the Basic and Hidden Server Architecture, G.726 audio codec provides the same audio quality than G.711 codecs.

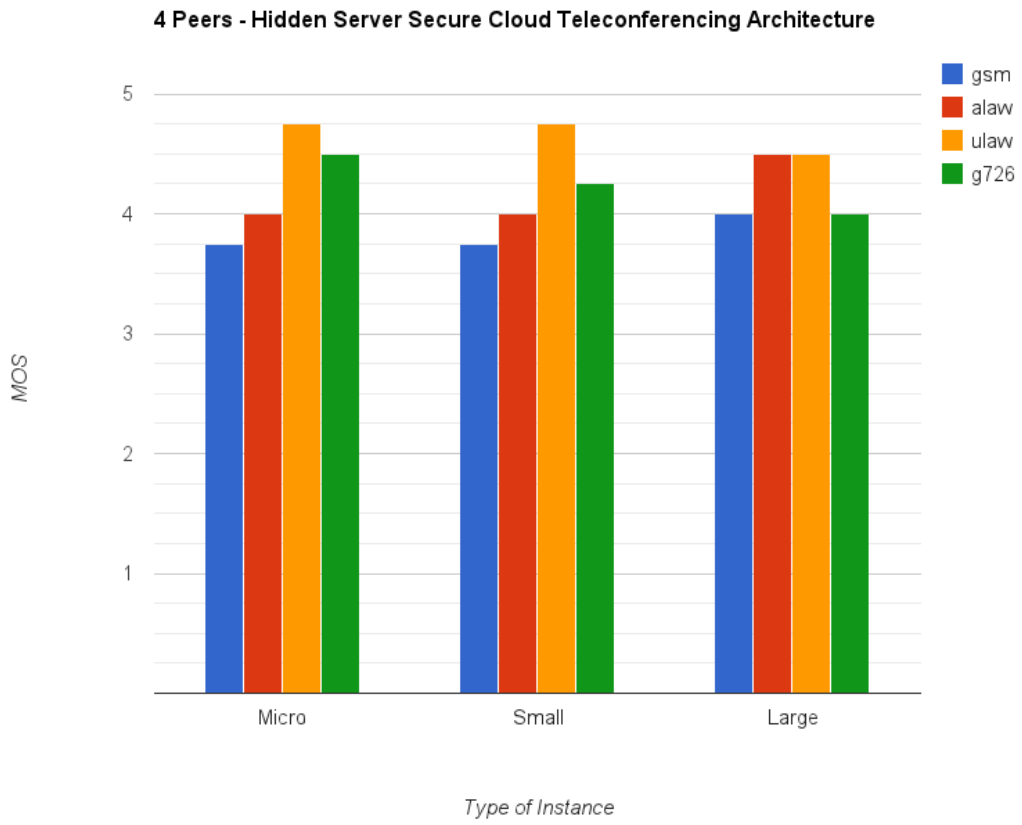

Figure 6.11: *Speech Quality in Hidden Server Secure Cloud Teleconferencing Architecture with 4 peers.*

6.1.3 6 peers Speech Quality

Micro Instances provides a significant less speech quality with all codecs when using 6 peers as shows Fig. 6.12 and Fig. 6.14 in comparison with Small and Large instances for all architectures. Indeed, in VPN architecture using Micro or Small Instances with 6 peers, quality degradation is very high, giving MOS values lower than fair quality (lower than three).

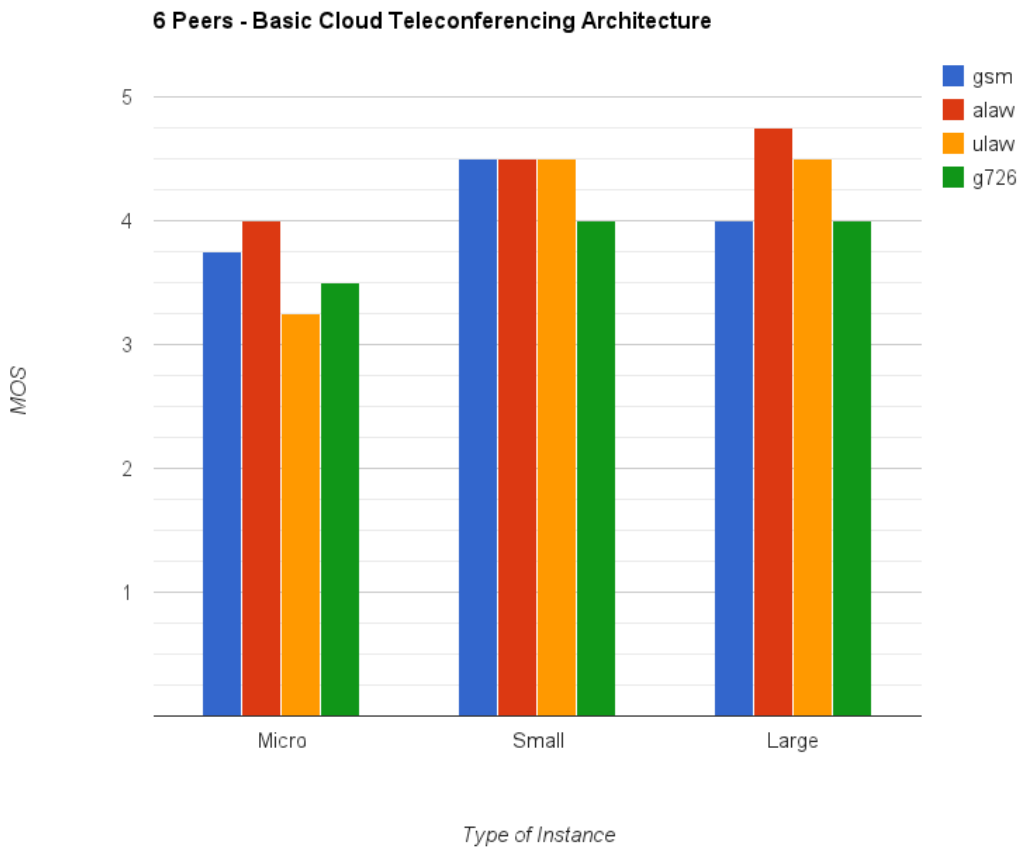

Figure 6.12: *Speech Quality in Basic Cloud Teleconferencing Architecture with 6 peers.*

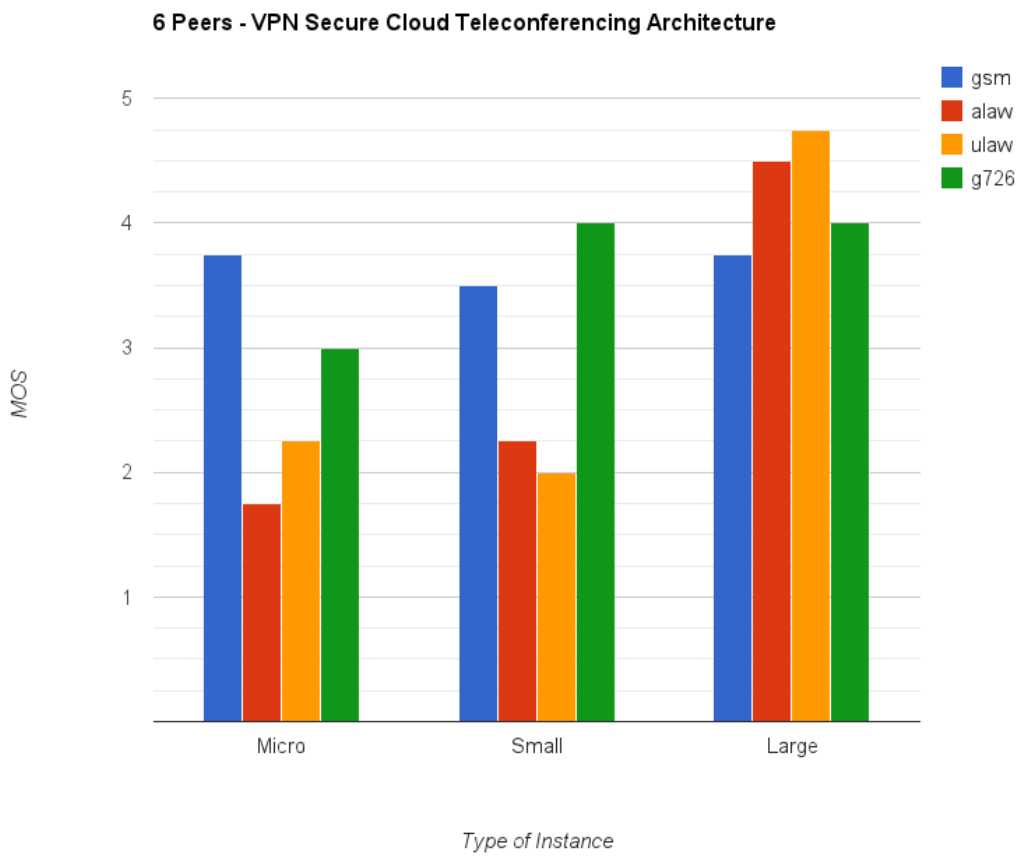

Figure 6.13: *Speech Quality in VPN Secure Cloud Teleconferencing Architecture with 6 peers.*

Unlike regular situations where G.711 codecs gives better MOS scores, when Micro and Small instances are overload in VPN architecture using 6 peers, GSM and G.726 audio codecs provide better results than G.711 audio codes [Fig 6.13].

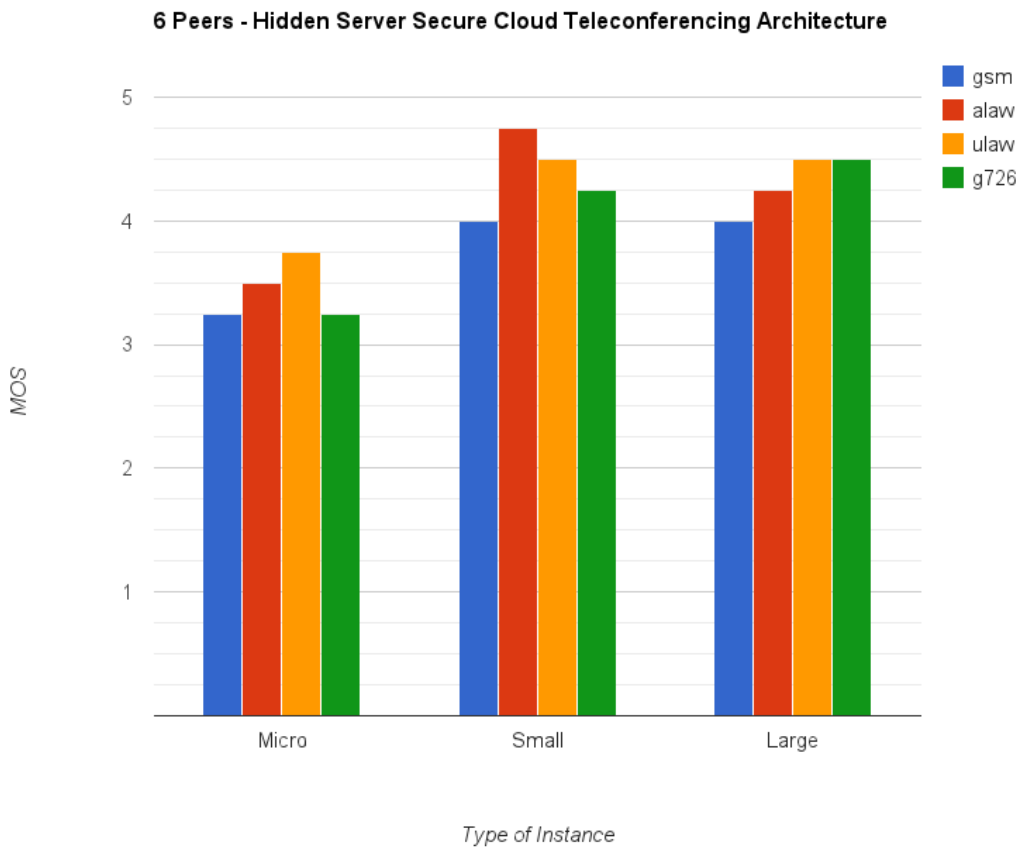

Figure 6.14: *Speech Quality in Hidden Server Secure Cloud Teleconferencing Architecture with 6 peers.*

6.1.4 8 peers Speech Quality

Speech quality considerably degrades in Micro and Small instances for all architectures when using 8 peers, being this degradation more significant than when using 6 peers. Large instances are not affected by connecting 8 peers, giving MOS scores similar to situations of 6, 4 and 2 peers.

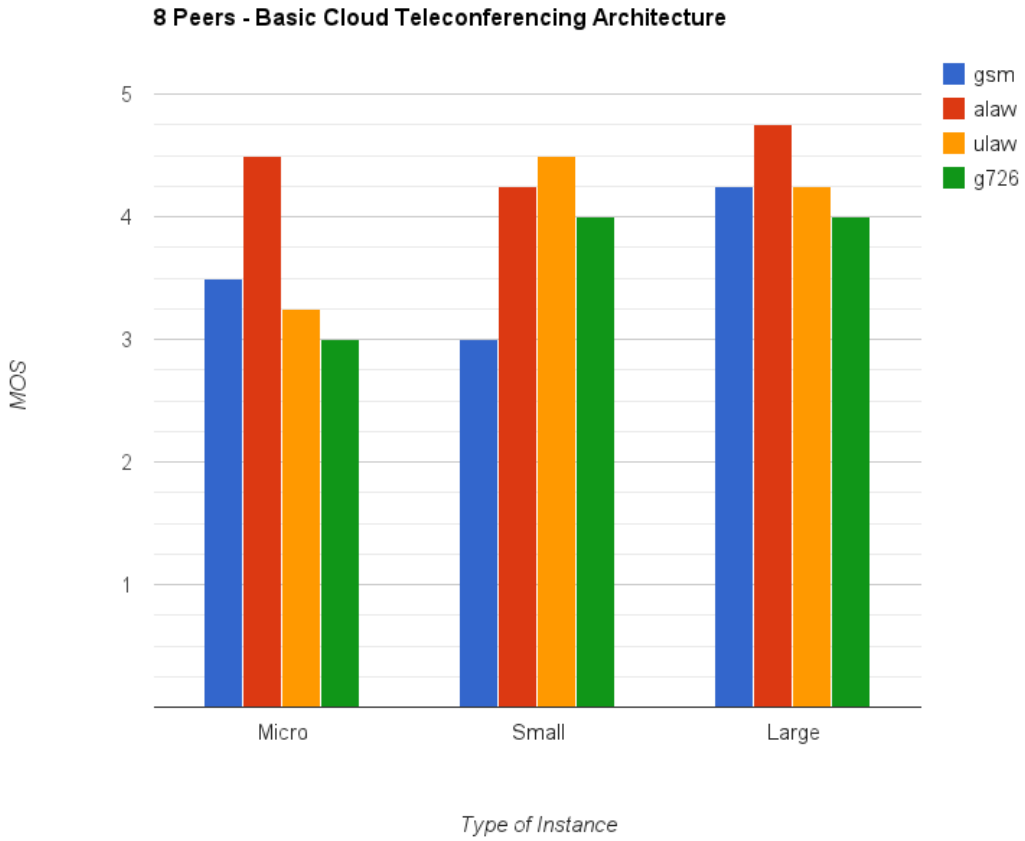

Figure 6.15: *Speech Quality in Basic Cloud Teleconferencing Architecture with 8 peers.*

As illustrated in Fig. 6.16, Micro and Small instances completely saturate using 8 peers in a VPN Architecture, hanging out as many as communications needed to overcome this saturation. Because of this, these results have not been included due to they are not significant to be cut some communications.

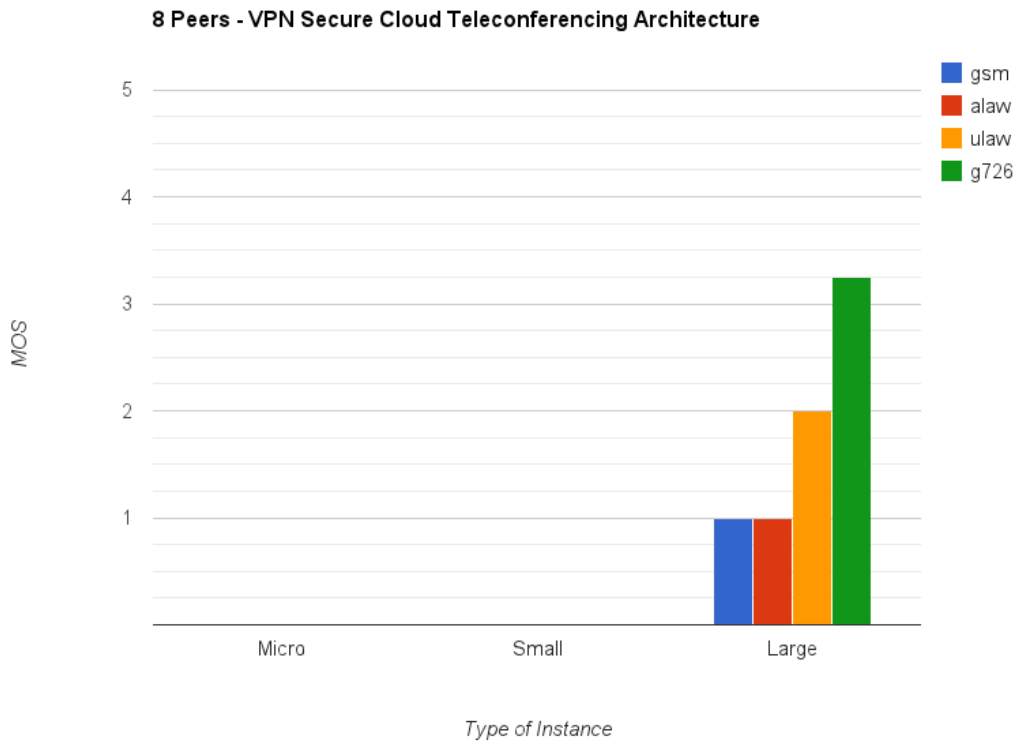

Figure 6.16: *Speech Quality in VPN Secure Cloud Teleconferencing Architecture with 8 peers. Micro and Small instances do not present results due to its saturation.*

G.711 audio codecs give better speech quality than the rest is most cases. However, in overload and saturation scenarios, G.726 audio codec gives same or better results than G.711 [Fig. 6.16 and Fig. 6.17].

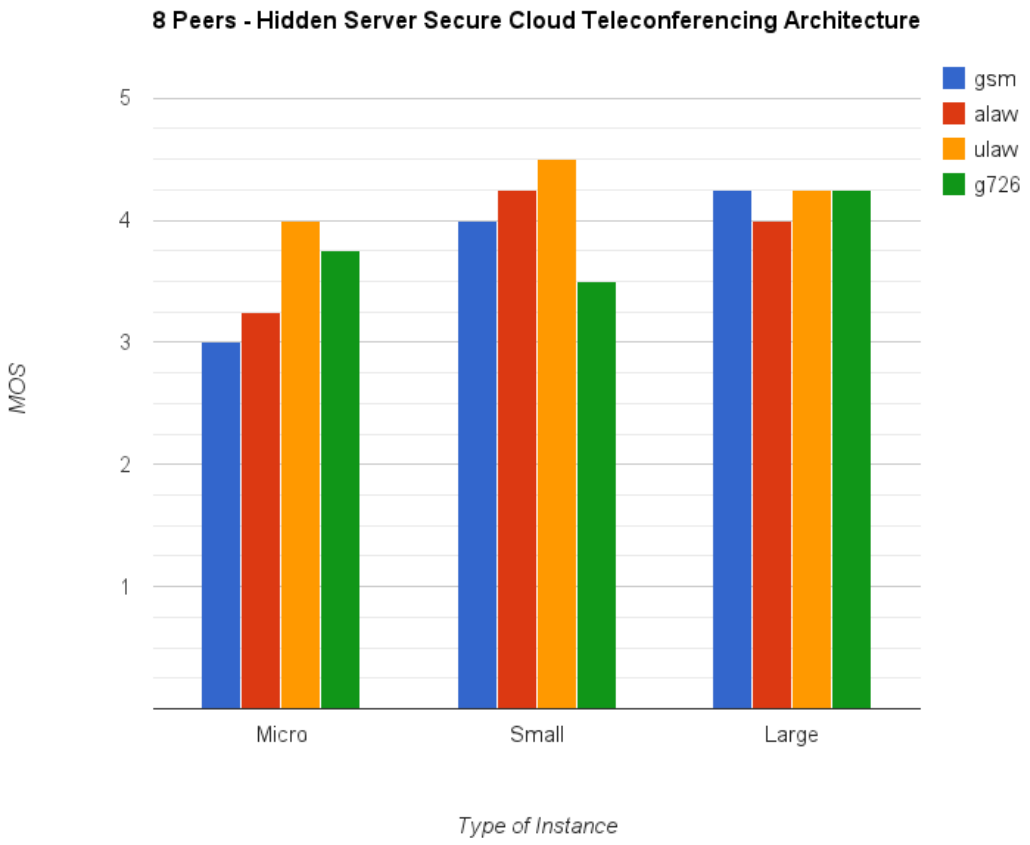

Figure 6.17: *Speech Quality in Hidden Server Secure Cloud Teleconferencing Architecture with 8 peers.*

6.1.5 Discussion

According to the obtained results, Large instances only provide better performance in saturated and overloaded scenarios, that is when using 6 or 8 peers. VPN architecture has lower performance than the others architectures, being likely due to the encryption and control flow extra load.

In general, audio codecs which gives better performance and speech quality are G.711 a-law and G.711 μ-law. Nevertheless, in overloaded and saturated scenarios, G.726 audio codec provides same or better results than G.711.

In regular scenarios (2 and 4 peers in Hidden Server Architecture), with neither overload nor saturation, instance type does not affect to the performance.

6.2 Deployment Cost-Speech Quality Results

It is proposed a new ratio to evaluate the performance against infrastructure costs. Infrastructure costs are in terms of price per kind of instance per hour. For the three selected instances hourly price is illustrated in Table 6.2.

Cost-Performance ratio is defined as follows:

$$R_{cp} = \frac{price\_per\_hour}{MOS}$$

Note that in Hidden Server Architecture, two Micro instances are launched for acting like End Points (firewalls), so in this architecture this cost needs to be added to the Asterisk server instance price.

For better comprehension and visual perception R_{cp} ratio has been multiplied by 100 in the following figures and illustrations.

R_{cp} values are presented by the number of peers for better comprehension. R_{cp} score is illustrated per each proposed architecture according each audio codec. The lower R_{cp} score, the better in terms of cost and its performance will the scenario be.

Type of Instance	Price per hour
Micro	$0,020
Small	$0,060
Large	$0,240

Table 6.2: *Price per hour for AWS selected instances in US-east zone for Linux OS.*

6.2.1 2 peers Deployment Cost-Speech Quality

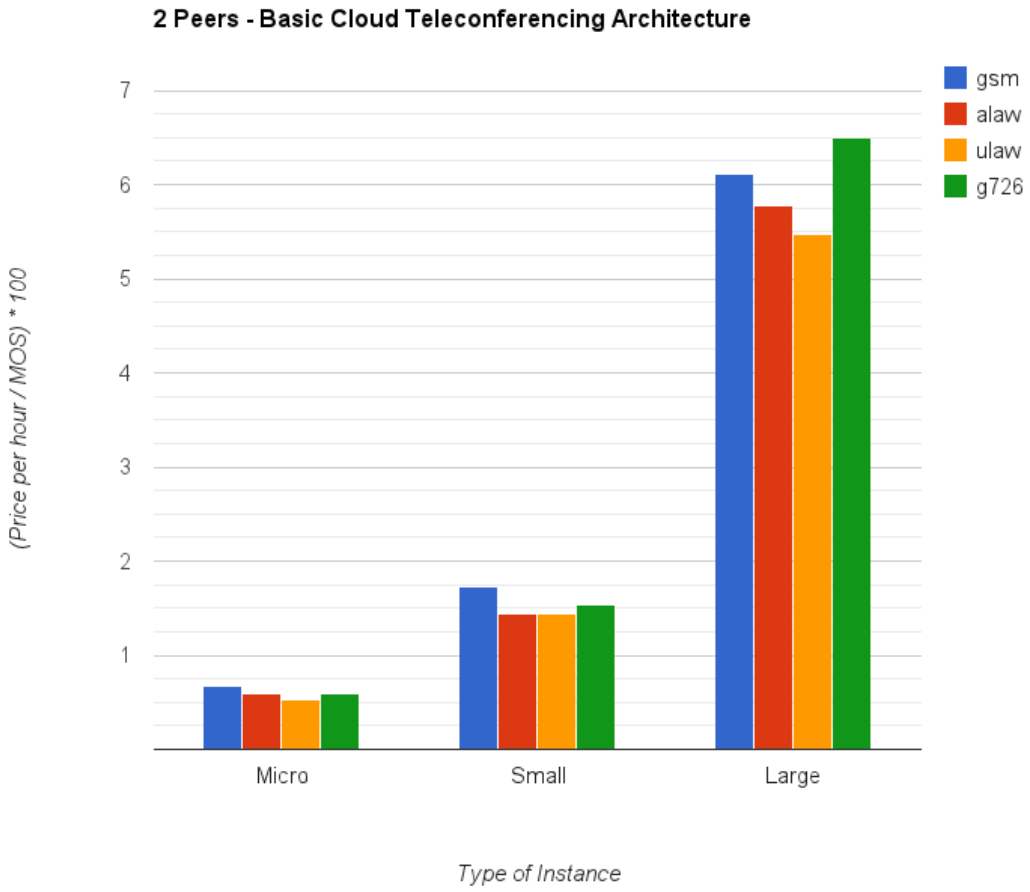

Figure 6.18: *R_{cp} for Basic Cloud Teleconferencing Architecture with 2 peers.*

Both in Basic as in VPN and Hidden Server Architecture, a better instance performance with its associated cost does not guarantee better speech quality results when using 2 peers [Fig. 6.18, Fig. 6.19 and Fig. 6.20]. In addition, in Large instances R_{cp} score soars, indicating that it is not worth spending money in Large instances comparing to the speech quality received.

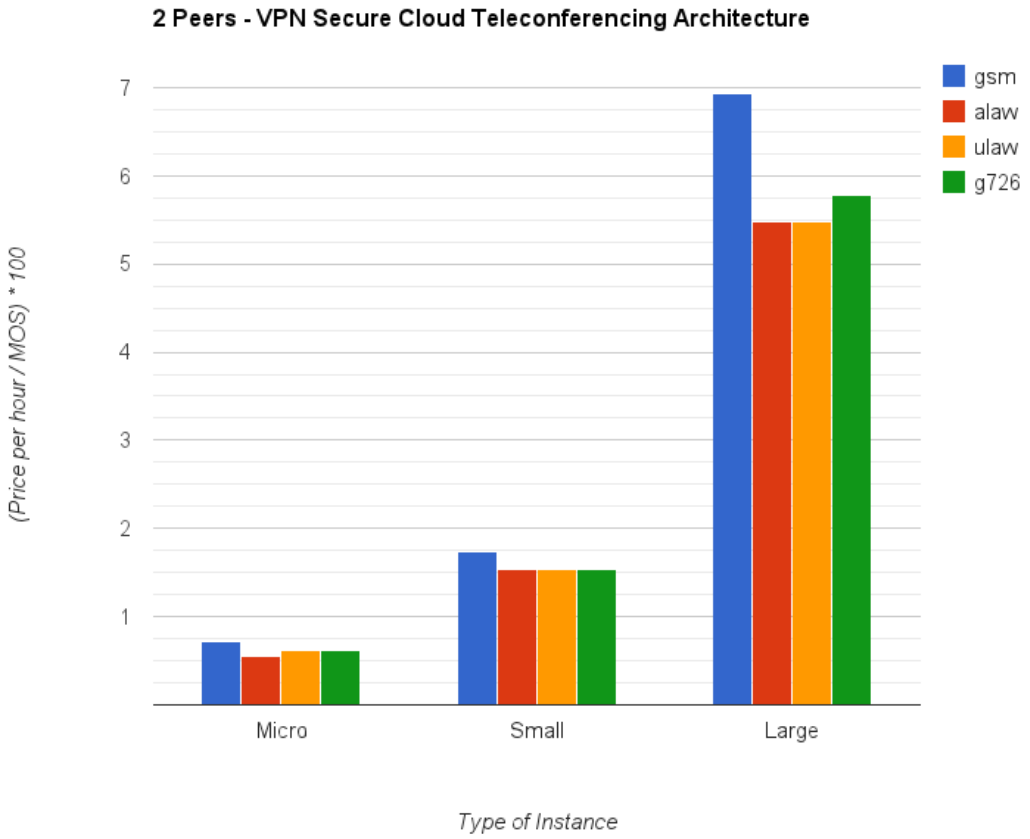

Figure 6.19: *R_{cp} for VPN Secure Cloud Teleconferencing Architecture with 2 peers.*

As is section 6.1.1, G.711 codecs provide better speech quality results, but also G.726 codec gives as good results. In a scenario with an Hidden Server Architecture deployed

and 2 peers, R_{cp} values raise because it is necessary to add two extra Micro instances (End Points) to deploy the whole infrastructure and this raises costs [Fig. 6.20].

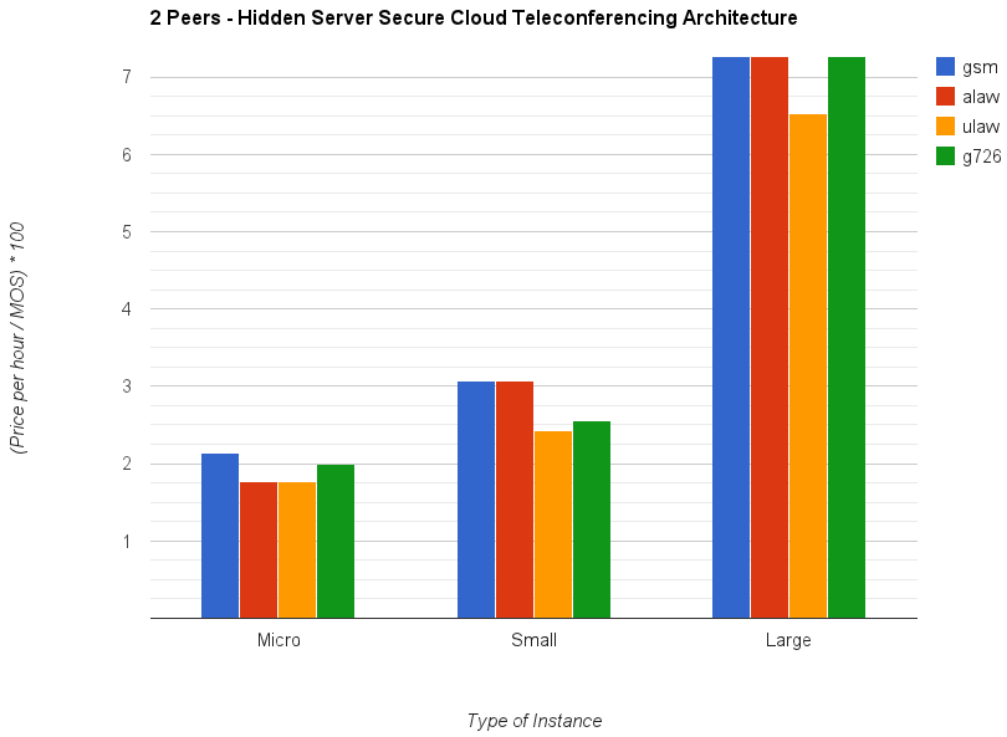

Figure 6.20: *R_{cp} for Hidden Server Secure Cloud Teleconferencing Architecture with 2 peers.*

6.2.2 4 peers Deployment Cost-Speech Quality

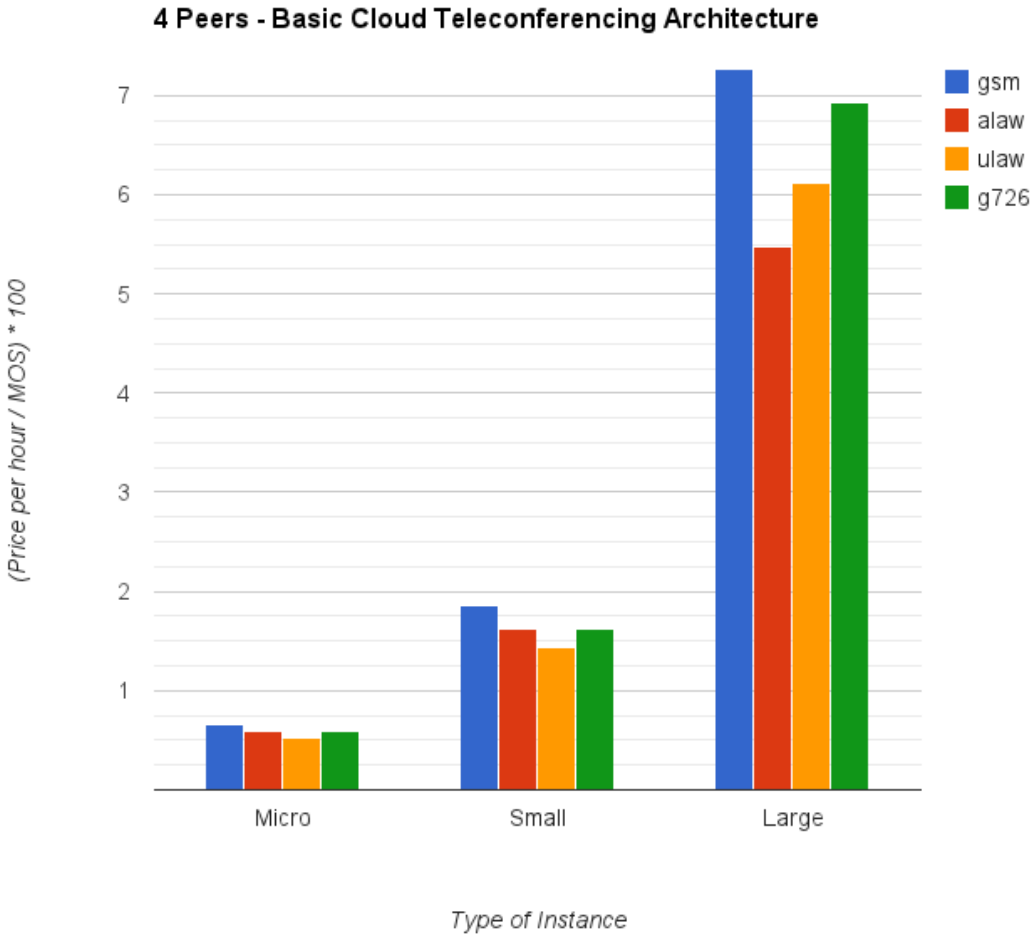

Figure 6.21: *R_{cp} for Basic Cloud Teleconferencing Architecture with 4 peers.*

In case of connecting 4 peers at the same time, Micro instances will be the best choice in terms of R_{cp} ratio since it is not worth spending more money in better instances as these do not guarantee better speech quality.

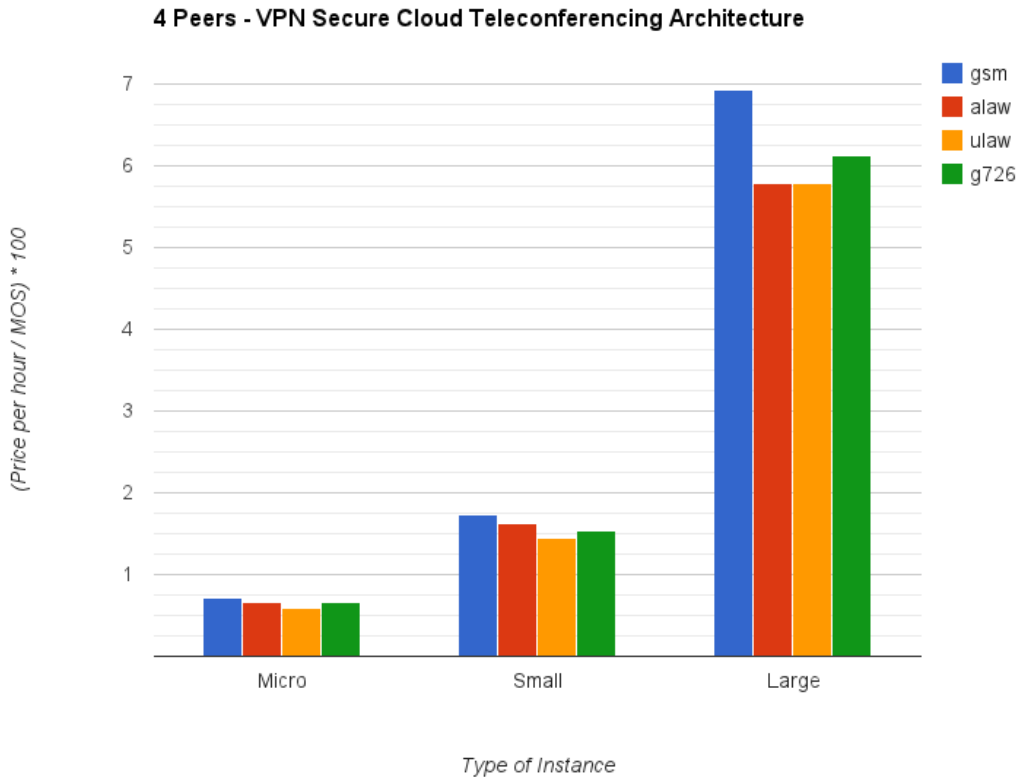

Figure 6.22: *R_{cp} for VPN Secure Cloud Teleconferencing Architecture with 4 peers.*

As in the previous case (connecting 2 peers), the best audio codec is G.711 μ-law in all the scenarios, giving better speech quality than the rest of the audio codecs with the same cost in each architecture. Again, R_{cp} values raise in Hidden Server Architecture due to the two extra instances cost.

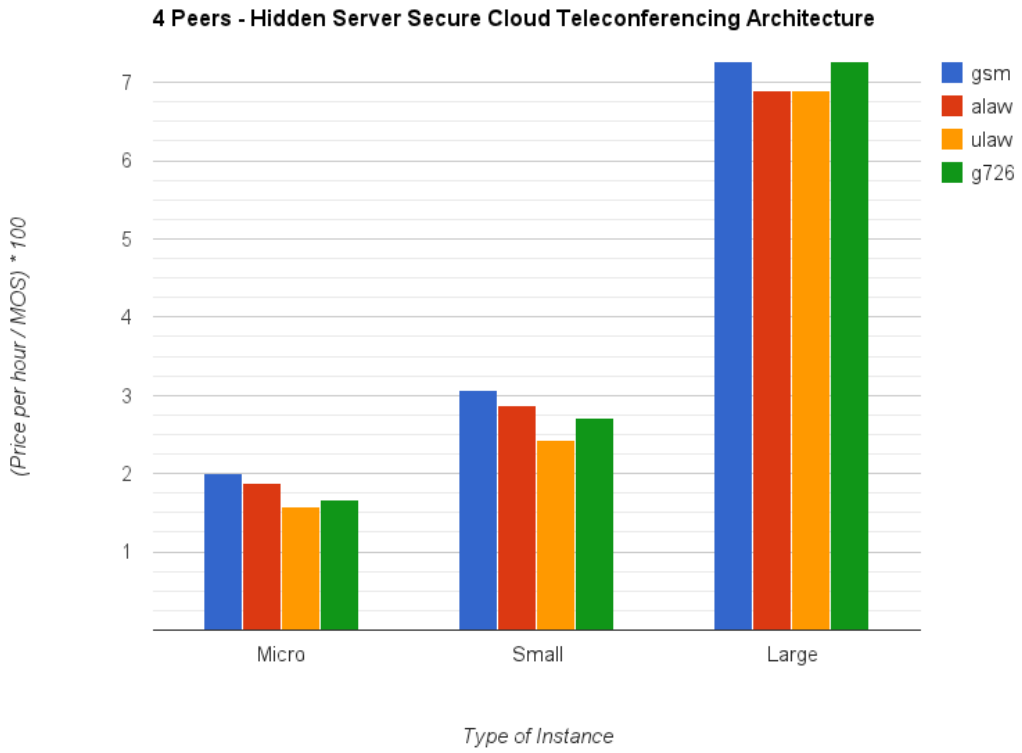

Figure 6.23: *R_{cp} for Hidden Server Secure Cloud Teleconferencing Architecture with 4 peers.*

6.2.3 6 peers Deployment Cost-Speech Quality

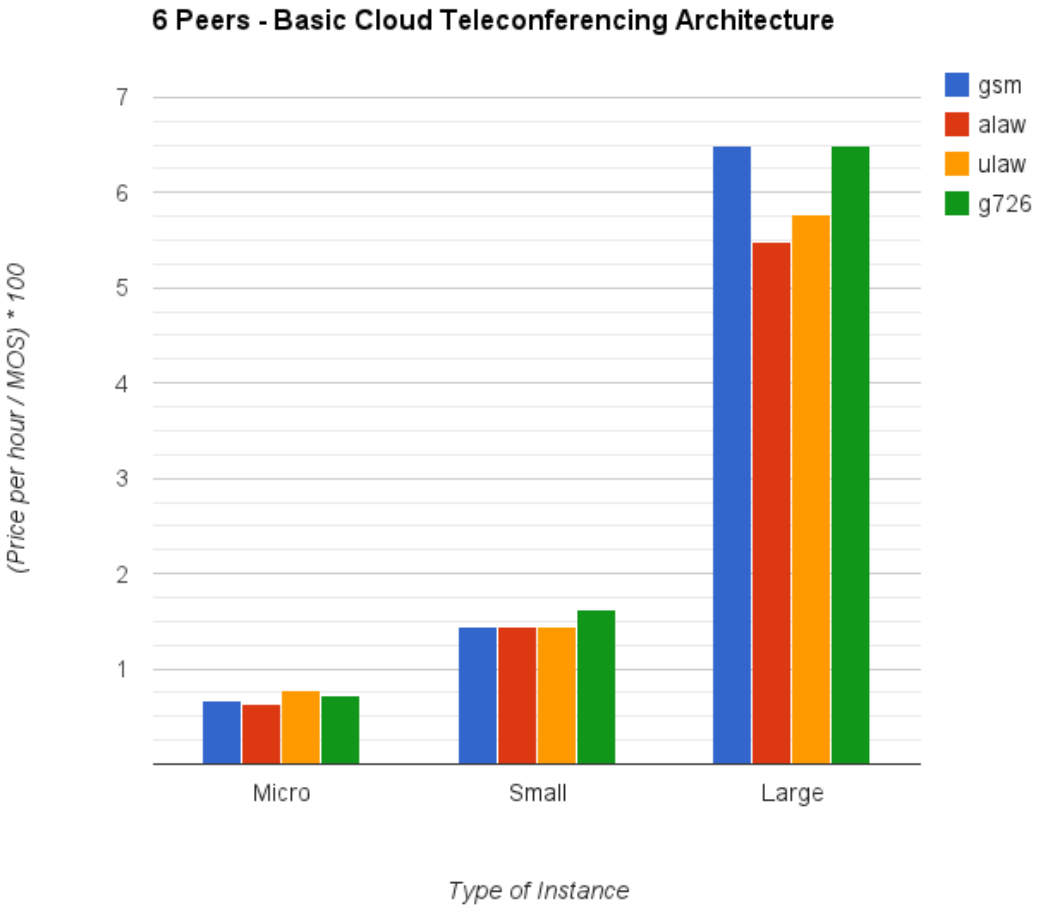

Figure 6.24: *R_{cp} for Basic Cloud Teleconferencing Architecture with 6 peers.*

VPN architecture gets overloaded when using 6 peers in Micro and Small instances [see section 6.1.3]. In Fig. 6.25 it is shown that differences between Micro instances R_{cp} scores and Small instances R_{cp} scores increase compared to the same scenario using 4 peers. System gets overloaded and provides same fair-low quality results for the two instances, not being worth spending more in acquire better instances.

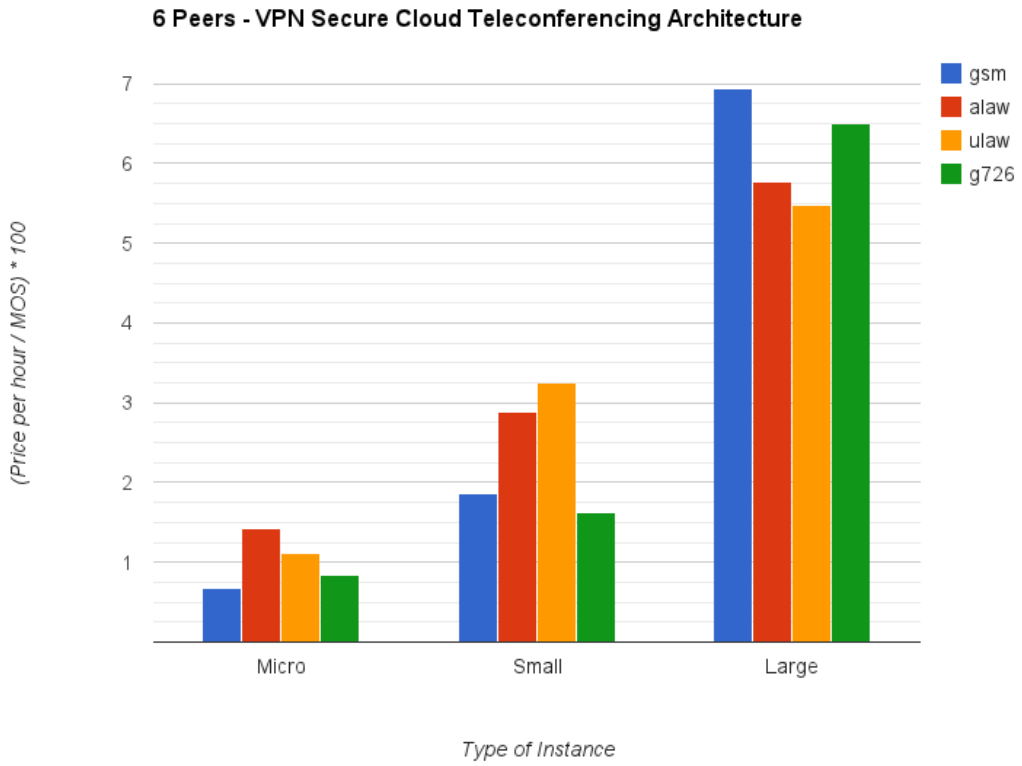

Figure 6.25: *R_{cp} for VPN Secure Cloud Teleconferencing Architecture with 6 peers.*

In Hidden Server Architecture, R_{cp} values of Micro and Small instances get closer [Fig. 6.26]. This is because enhancing instance features in this kind of architecture connecting 6 peers, a better quality is received. G.711 μ-law audio codec in regular conditions and G.726 in overload conditions [Micro and Small instances in Fig. 6.25], provide better speech quality with the same cost than the rest of codecs.

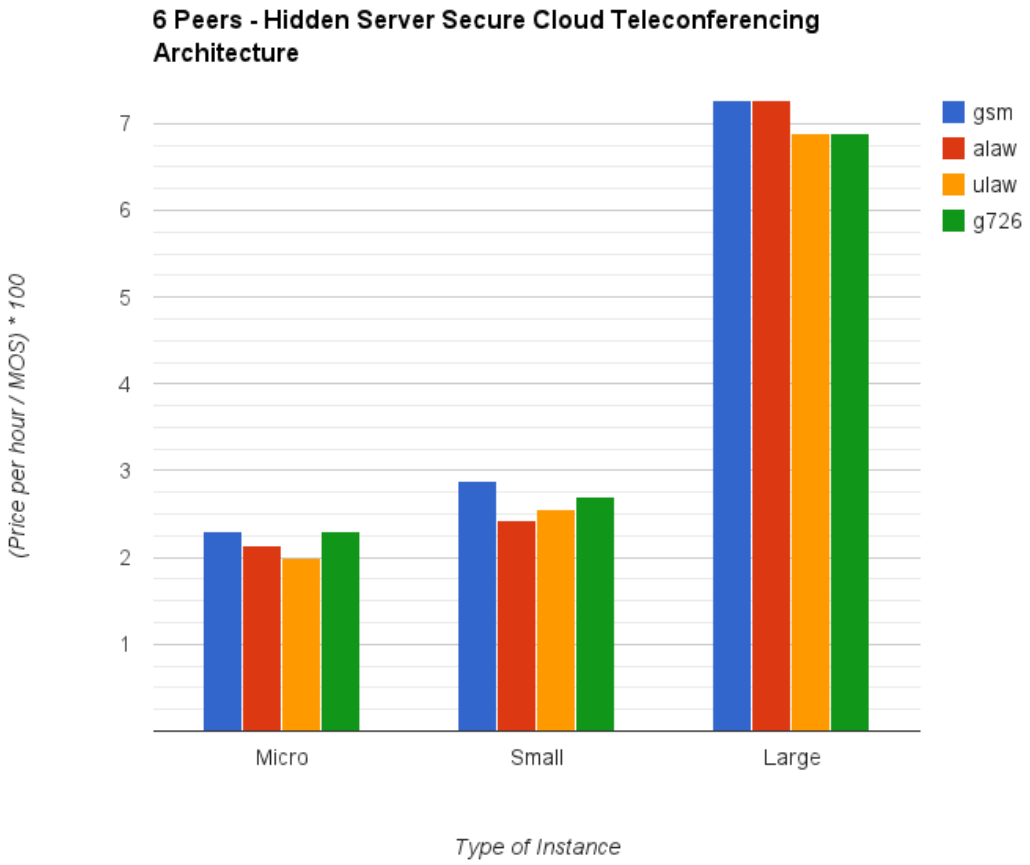

Figure 6.26: *R_{cp} for Hidden Server Secure Cloud Teleconferencing Architecture with 6 peers.*

6.2.4 8 peers Deployment Cost-Speech Quality

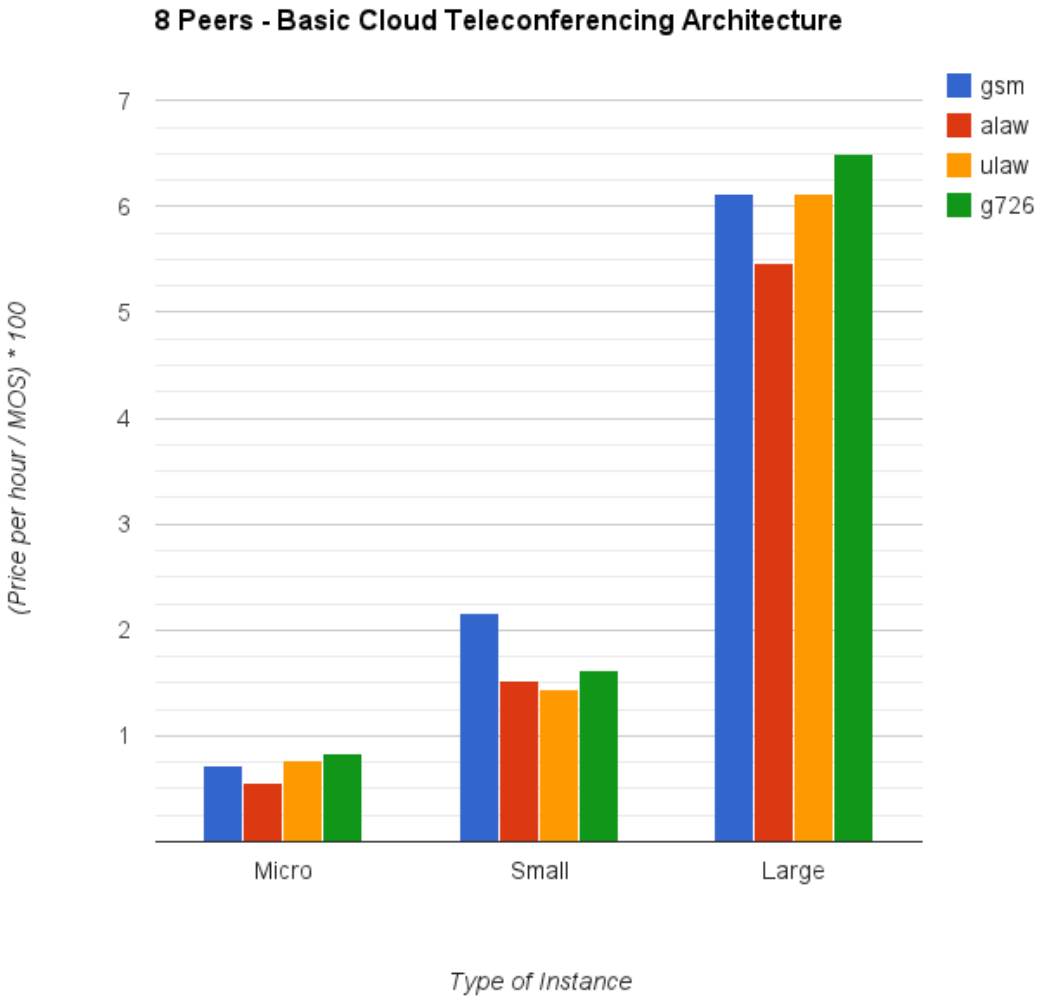

Figure 6.27: *R_{cp} for Basic Cloud Teleconferencing Architecture with 8 peers.*

As in the previous scenarios, Micro instances continue being the best choice in terms of cost per quality of service [Fig. 6.27 and Fig. 6.29]. However, in Hidden Server Architecture, as happened connecting 6 peers, R_{cp} scores get closer because better speech quality is obtained

when instance features are enhanced.

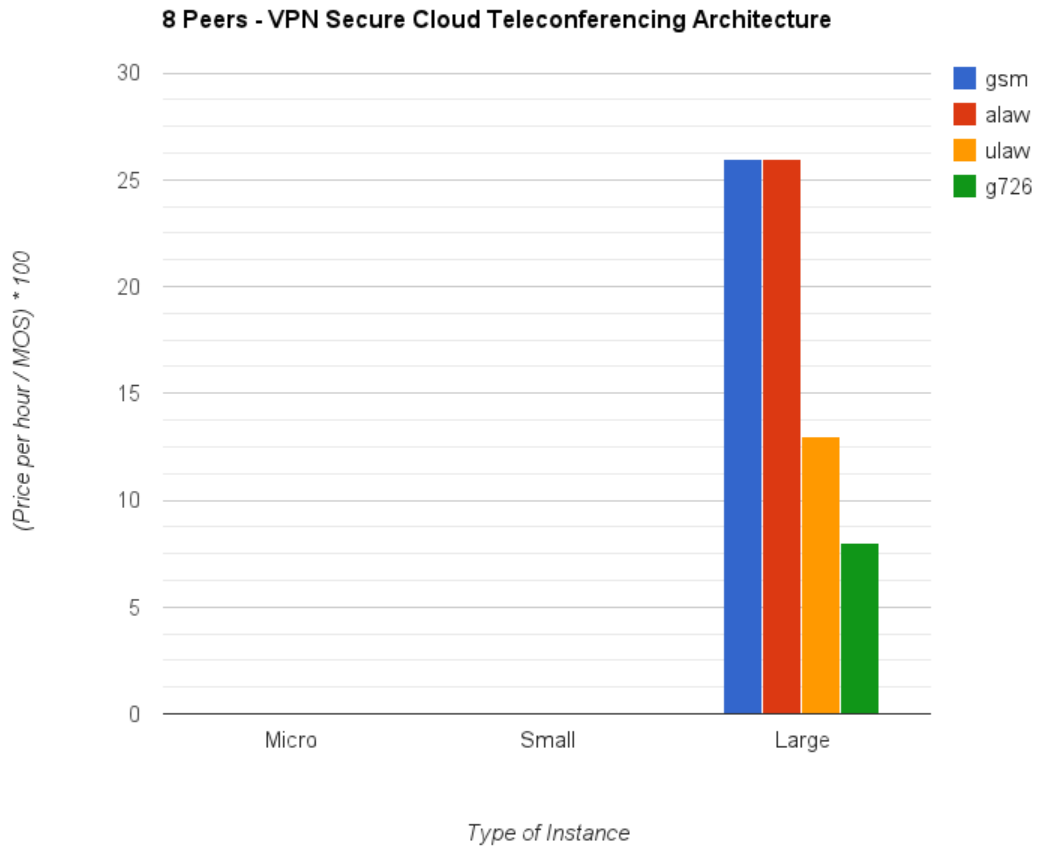

Figure 6.28: *R_{cp} for VPN Secure Cloud Teleconferencing Architecture with 8 peers.*

R_{cp} score are not presented in Fig. 6.28 in case of connecting 8 peers to VPN architectures due to system saturation with Micro and Small instances, so in this case, the only possibility would be acquire a Large instance. In contrast with the previous results in high degradation scenarios as in Fig. 6.28, G.726 audio codec gives better speech quality results for the same deployment costs.

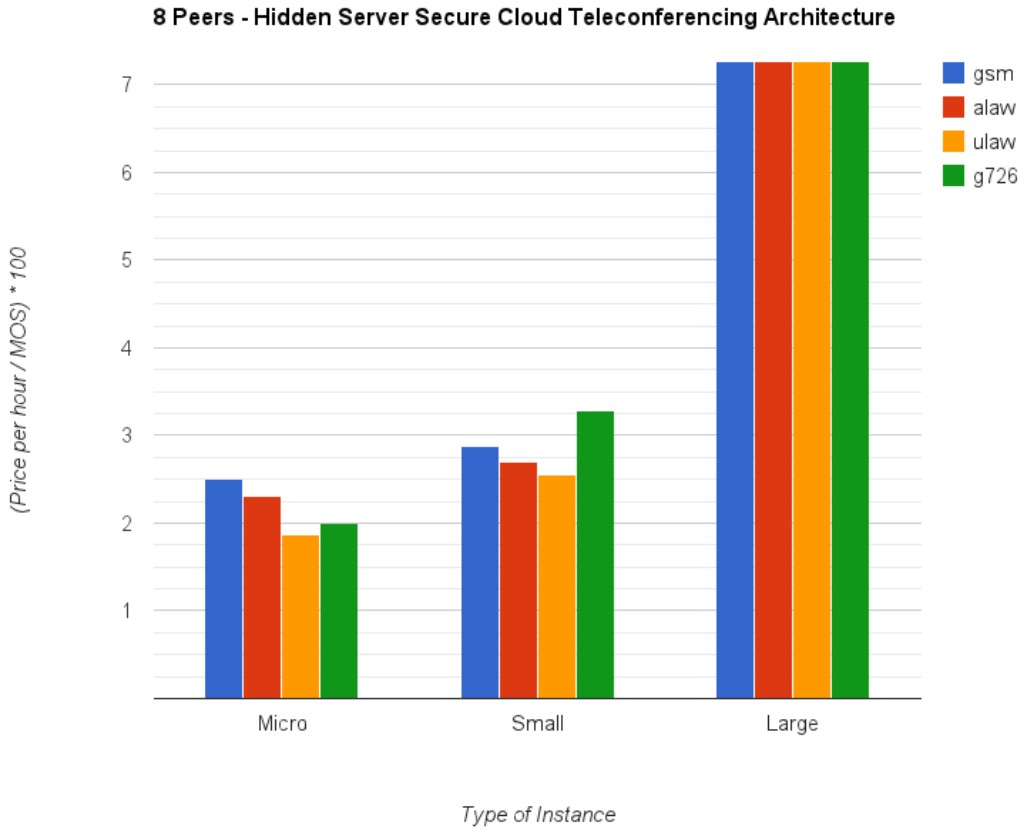

Figure 6.29: *R_{cp} for Hidden Server Secure Cloud Teleconferencing Architecture with 8 peers.*

6.2.5 Discussion

In a global outline according to the new established metric R_{cp}, Micro Instances are much cheaper, providing similar quality of service and performance than Small and Large Instances in scenarios with 2 or 4 peers. In scenarios with 6 peers, depending on the architecture, a Micro or Small instance would be a better choice. For example, for Hidden Server Architecture with 6 peers, Small instance would be suitable instead of Micro instance if quality is much important than cost. Finally, for 8 peers situations, neither Micro nor Small instances

can establish calls in VPN architectures, being necessary to use Large instances. However, when deploying Hidden Server Architecture with 8 peers, as happened with 6 peers, Small instances would be a better choice than Micro instances if quality of service is more important than deployment costs.

Regarding the audio codecs, G.711 codecs give better speech quality for the same scenario than G.726 and GSM for non overloaded and saturated systems. On the other hand, G.726 and GSM codecs are better for overloaded and saturated situations as illustrated in Fig. 6.25.

Chapter 7

Model

According to the results and the later discussion, an infrastructure decision tree can be obtained based on user preferences [Fig. 7.1]. This model shows a decision tree whose steps are in the same way as results were illustrated: number of peers, kind of architecture, type of instance and finally, audio codec. Lower than 3.5 MOS values would be consider a non acceptable quality of service.

The infrastructure decision tree flow goes as follows:

1. The decision process starts in determine the number of peers that will be used within the secure cloud teleconferencing architecture.

2. Once the number of peers is fixed, user needs to decide what it is more important security or performance, and establish if peers are in a secure environment. When these decisions are taken, Secure Cloud Architecture is set, choosing between VPN or Hidden Server Secure Cloud Teleconferencing Architecture.

3. After fixing the number of peers and the architecture, the type of instance is set based on the cost, performance or both (R_{cp}).

4. Audio codec is automatically set after the three previous steps, fixing the audio codec that provides better speech quality for the scenario.

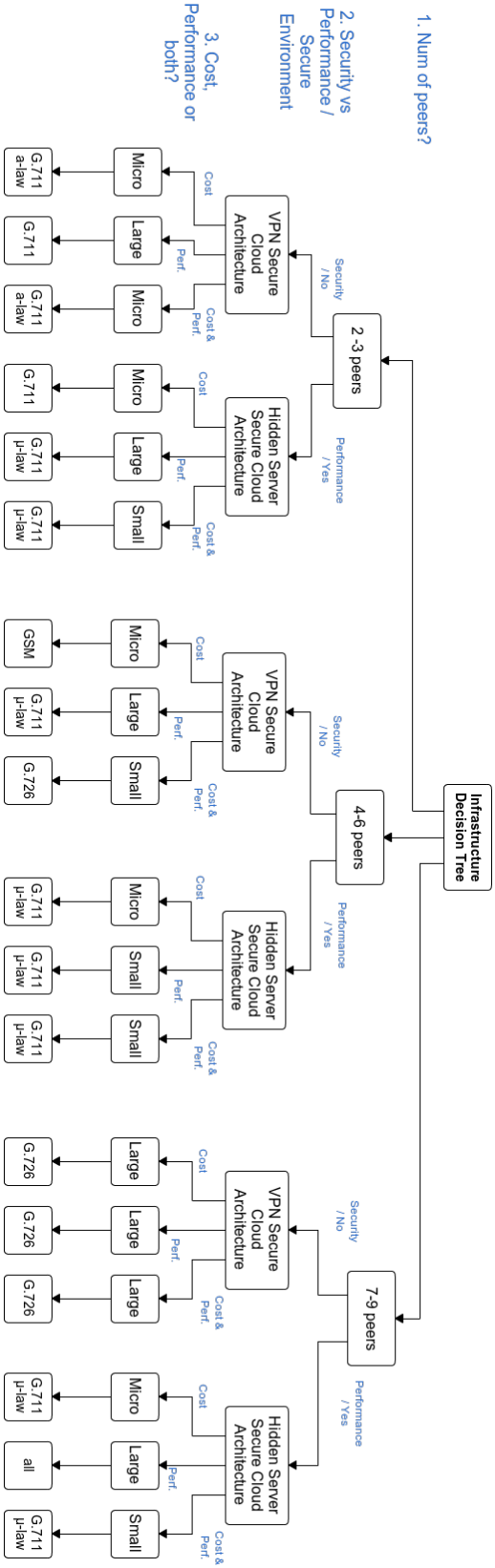

Figure 7.1: *Secure Cloud Infrastructure decision tree.*

With this model, an user who wants to deploy a secure cloud teleconferencing infrastructure can easily decide which is the best option for him just answering a couple of basic questions that will determine the most suitable codec, security and architecture according to his necessities.

Part III

Conclusions and future work

Chapter 8

Contributions

This chapter discusses what are the main contributions of the project and draws some conclusions about the work carried out.

In accordance to the correlation results between SNR, PNR and PESQ with MOS, the only acceptable VoIP QoS estimator is PESQ with a correlation of nearly 0.78. On the other hand, SNR and PSNR are not valid QoS estimators, both with an extremely poor correlation around 0.07. These behaviours are similar to the results and conclusions achieved in previous works[41][3][42][26][6][38].

Network latency does not affect in a uniform way to speech quality (a higher delay does not imply lower quality), although more low MOS scores are given with high delay. Moreover, delay impact is higher in VPN Secure Cloud Teleconferencing Architecture. Despite the maximum point-to-point delay in VoIP communications defined by ITU (between a preferred delay of 150ms and a maximum of 400ms) [Table 2.2], higher latency values continue getting high quality results (more than 3.75 MOS values).

The minimum MOS value QoS for the proposed architectures is set to 3.5. Other VoIP applications such as Skype or GTalk do not exceed a 3.75 MOS value, even with excellent network conditions with a delay lower than 50 ms [33][27][44]. The proposed architectures,

in spite of the security measures overload, provide higher QoS, reaching excellent speech quality levels above 4 MOS scores. Although Skype and GTalk (now Google+ Hangouts) have millions of users, the proposed secure architectures could adopt and elastic and distributed infrastructure in the cloud to provide service to users depending on demand.

The main contribution of this work is the proposal of two Secure Cloud Teleconferencing Architectures, VPN Secure Cloud Teleconferencing Architecture and Hidden Server Secure Cloud Teleconferencing Architecture, to cover security issues in public clouds and Internet Communications. These architectures define which security model, cloud instances infrastructure and audio codec to use to achieve the best performance while preventing of security breaches and the risks associated to VoIP. VPN Secure Cloud Teleconferencing Architecture can prevent from registration, on call and denial of service attacks. On the other hand, Hidden Server Secure Cloud Teleconferencing Architecture can prevent from denial of service attacks and attacks to VoIP main component, Asterisk server.

These proposed architectures are reflected in a secure cloud teleconferencing infrastructure decision tree, where users who need to deploy a secure cloud VoIP architecture can easily choice the best option according to their preferences in terms of cost, security and performance. This model determines from the user's answers to three questions, the most suitable secure architecture, type of AWS to use and the best audio codec to achieve better QoS.

Now, all efforts can be focused on the Asterisk server elasticity to fully validate the proposed architectures and the model.

Chapter 9

Future work

The proposed Secure Cloud Teleconferencing Architectures solve security breaches in public cloud and Internet communications addressing the concept of security in VoIP based on three fundamentals: Confidentiality, Integrity and Availability. Each proposed architecture solves some of the VoIP attacks identified but not all, being essential a security improvement to deal with all possible VoIP attacks. Future work from a security perspective includes:

- **A combination between VPN and Hidden Server Secure Cloud Teleconferencing Architecture**. VPN Architecture prevents from registration, on call and denial of service attacks and Hidden Server Architecture prevents from denial of service attacks and attacks on VoIP Server. A combination between these two architectures would prevent from the four attacks [see section 2.2]. This new architecture has not been addressed in this work due to the problems related to NAT and SIP protocol.

- Designing **other secure architectures with different kinds of encryption** and/or authentication and how affects to QoS. For example, RSA for asymmetric cyphering will add extra computation time to OpenVPN server but will not have to deal with control flow of VPN architectures. On the other hand, symmetric cyphering methods like AES, DES or TDEA will improve computation time of encryption/decryption, but are less secure and more susceptible to be attacked.

On the other hand, from an infrastructure perspective, cloud secure architectures need to evolve including some of the following features:

- **An scalable and distributed VoIP server**. Deploying an scalable and distributed Asterisk server would provide a better QoS and save costs, adapting on demand the Asterisk servers to the needs of the moment.
- **On demand security and deployment**. An architecture that differentiates security according to network infrastructure and application would avoid an unnecessary drain on IT resources by protecting each communication at just the right level[24]. Moreover, an architecture that is able to deploy end points and VoIP server in the nearest AWS zones to clients would improve latency issues.

Furthermore, including other kind of wireless and mobile networks (such as 3G and the 4G) as well as including a higher number of peers will enrich this study and the secure cloud teleconferencing infrastructure decision tree.

Bibliography

[1] Voip wiki - a reference guide to all things voip. http://www.voip-info.org.

[2] Cert vulnerability note vu836068. Kb.cert.org, 2010.

[3] F.P. Freeland et al. A. A. de Lima. On the quality assessment of sound signals. *IEEE*, 2008.

[4] E. Ackermann and K. Hartman. *Internet and Web Essentials: What You Need to Know*. Wilsonville OR, 2000.

[5] Skype and/or Microsoft. Skype. http://www.itu.int/rec/T-REC-P.862-200102-I/en.

[6] Anil Kokaram Andrew Hines, Jan Skoglund and Naomi Hartez. Robustness of speech quality metrics to background noise and network degradations: comparing visqol, pesq and polga. *IEEE*, 2013.

[7] Teri Bidwell. *Hack Proofing Your Identity in the Information Age*. Syngress Publishing, 2002.

[8] Philippe Biondi and Fabrice Desclaux. Silver needle in the skype. *Black Hat Europe*, 2006.

[9] R. Braden. Rfc: 1122. requirements for internet hosts – communication layers. *Internet Engineering Task Force*, 1989.

[10] R. Braden. Rfc: 1123. requirements for internet hosts – application and support. *Internet Engineering Task Force*, 1989.

[11] Hsiao-Hwa Chen and Hamid R. Sharif. *Security and Communication Networks*. John Wiley & Sons, Ltd, 2011.

[12] CompuBase Consulting. Challenges & opportunities for it partners when transforming or creating a business in the cloud, 2012.

[13] Microsoft Corporation. Protección y privacidad en línea. http://www.microsoft.com/es-es/security/online-privacy/passwords-create.aspx.

[14] Inc Digium. Asterisk. http://www.asterisk.org/.

[15] Electronic Frontier Foundation. *Cracking DES - Secrets of Encryption Research, Wiretap Politics & Chip Design*. Oreilly & Associates Inc, 1998.

[16] et al. Greene. Rfc: 2805. mg control protocol requirements. *Internet Engineering Task Force*, 2000.

[17] Peter H. Gregory. *VoIP Security for Dummies*. Wiley Publishing, Inc., 2006.

[18] Alexa Huth and James Cebula. The basics of cloud computing. *US-CERT*, 2011.

[19] Google Inc. How safe is your password? https://accounts.google.com/PasswordHelp.

[20] ITU. Methods for subjective determination of transmission quality. *TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU*, 1996.

[21] ITU-T. P.862 : Perceptual evaluation of speech quality (pesq): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. http://www.itu.int/rec/T-REC-P.862-200102-I/en.

[22] ITU-T. Función de correspondencia para convertir los resultados brutos de la prueba p.862 en nota media de opinión de la calidad de escucha objetiva. *SECTOR DE NORMALIZACIÓN DE LAS TELECOMUNICACIONES DE LA UIT*, 2003.

[23] ITU-T. The e-model: a computational model for use in transmission planning. *TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU*, 2011.

[24] Yang Wang Jianyong Chen and Xiaomin Wang. On-demand security architecture for cloud computing. *IEEE Computer Society*, 2012.

[25] Brian Krebs. Privacy 101: Skype leaks your location. *Krebs on Security*, 2013.

[26] Philipos C. Loizou. Speech quality assessment. 2011.

[27] Álvaro Masó Marc Cardenete-Suriol, Josep Mangues-Bafalluy and Mónica Gorricho. Characterization and comparison of skype behavior in wired and wireless network scenarios. *Proc. IEEE Globecom*, 2007.

[28] Peter Mell and Timothy Grance. The nist definition of cloud computing. *National Institute of Standards and Technology*, 2011.

[29] Jeffrey Bosma Michiel Appelman and Gerrie Veerman. Viber communication security. 2011.

[30] Inc. OpenVPN Technologies. Openvpn security overview. http://openvpn.net/index.php/open-source/documentation/security-overview.html.

[31] Diego Guerra Vidal y Ignacio Irigaray Bayarres Pedro Casas Hernández. Calidad de servicio percibida en servicios de voz y video sobre ip. Final degree project, Universidad de la República, 2005.

[32] S.M.A. Rivzi and P.S. Dowland. *Advances in Networks, Computing and Communications 4*, chapter VoIP Security Threats and Vulnerabilities, pages 114–122. University of Plymouth, UK, 2005.

[33] Dênio Mariz et al. Rodrigo Barbosa, Carlos Kamienski. Performance evaluation of p2p voip applications. *NOSSDAV, Urbana, Illinois USA.*, 2007.

[34] Bruce Schneier. Schneier on security: Cryptanalysis of sha-1, 2005.

[35] Bruce Schneier. Choosing secure passwords, 2007.

[36] Amazon Web Services™. Amazon elastic compute cloud user guide. http://awsdocs.s3.amazonaws.com/EC2/latest/ec2-ug.pdf.

[37] Cisco Systems. Understanding codecs: Complexity, hardware support, mos, and negotiation. http://www.cisco.com/en/US/tech/tk1077/technologies_tech_note09186a00800b6710.shtml, 2006.

[38] Mindspeed Technologies. Measuring voice quality. 2002.

[39] Jan Kanclirz Jr. Thomas Porter and Brian Baskin. *Practical VoIP Security*. Andrew Williams, 2011.

[40] R. Venkateswaran. Virtual private networks. *IEEE Potentials*, 2011.

[41] Stephen D. Voran. Objective estimation of perceived speech quality using measuring normalizing blocks. *NTIA Report 98-347*, 1998.

[42] Stephen D. Voran. Objective estimation of perceived speech quality, part ii: Evaluation of the measuring normalizing block technique. *IEEE Transactions on Speech Audio Processing*, 1999.

[43] Xiaoyun Wang and Hongbo Yu. How to break md5 and other hash functions. *EUROCRYPT'05 Proceedings*, 2005.

[44] Wei-Cheng Xiao Wen-Hui Chiang and Cheng-Fu Chou. A performance study of voip applications: Msn vs. skype. *MULTICOMM*, 2006.

[45] Kamil Wojcicki. Pesq matlab wrapper. http://www.mathworks.es/matlabcentral/fileexchange/33820-pesq-matlab-wrapper.

[46] Shiping Chen Xinyuan Wang and Sushil Jajodia. Tracking anonymous peer-to-peer voip calls on the internet. *CCS*, 2005.